Inverse Modelling Requirements for a Nuclear Materials Safeguards Tool

By

Euan Colin Miller
B.Eng (Hons)

A thesis presented to the University of Glasgow, in fulfilment of the requirements of the degree of Doctor of Philosophy

Centre for Systems and Control
Department of Mechanical Engineering
University of Glasgow

Research Supervisor: Dr J. Howell

For Jane

ABSTRACT

The work presented in this thesis has been carried out in the support of the specification of a solution monitoring system to assist United Nations' inspectors performing nuclear materials, primarily pertaining to the chemical separation areas of nuclear reprocessing facilities. The system is designed to provide assurances over hours and days, other methods are more appropriate for the provision of assurances over weeks. The impetus for this system derives from the fact that conventional material accountancy methods are unable to satisfy the protracted loss detection goal specified by the International Atomic Energy Agency when applied to large commercial reprocessing plants.

Based on the concept of model-based reasoning, the system estimates the distribution of plutonium throughout the plant via simulation, and then attempts to justify any discrepancies between the estimated distribution and the observed distribution. Because the simulation's structure is fixed the process of justification involves hypothesising additional forcing functions and parameter changes, which result in the simulation predicting that observed. The simulation inputs are largely in the form of flow rates and concentrations, which are obtained via indirect measurement.

Plant operators discourage invasive measurement systems on the grounds of the expense of maintenance and plant containment. For this reason the direct measurement of material flow rates is not possible. However, the volume and density of liquor in process tanks is measured, so it is possible to obtain the flow rates indirectly by analysing the measurements, a process known as inverse modelling. Concentration measurements are obtained from the laboratory analysis of samples.

Inverse modelling is not just confined to flow rate estimation, because one of the aims of the system is also one of inverse modelling: to hypothesis a set of forcing functions and boundary conditions which, when input into the simulation, predicts the observed distribution. Thus inverse modelling is required at two levels, locally for flow rate estimation and more globally for distribution estimation over the entire plant.

Inverse modelling is problematic because inverse solutions have a propensity to be non-unique and unstable. Furthermore, since the solutions are obtained by analysing the measurements, they are adversely affected by the presence of noise and/or biases. This thesis describes some of the tools that have been developed as part of this system. A number are based on common statistical process control algorithms such as the Shewhart Control Chart and the V-mask, others involve more novel algorithms such as simulated annealing. Different tools are used over different time-scales: the short-term and the medium-term

Over the short-term, disagreements between the simulation and observations are analysed to generate forcing function hypotheses by using banks of observers to generate a list of the possible causes. The most likely hypothesis is chosen on the basis of user specified subjective possibilities. These probabilities reflect the view that some events are more likely to be acceptable to the operator than others are.

The problem over the medium-term is more difficult. The inverse modelling process is imperfect so the model diverges from the real plant over time with the net effect that quantities of material are predicted to be in the wrong place. This imperfection can stem from both the simulation and the plant data. The possible causes are biases that may exist on the plant and inaccuracies in the estimation of flow rates that affect the simulation. A method is proposed for identifying and estimating the gross multiplicative biases. If no bias is found an event is created describing the redistribution necessary to achieve parity. A method is proposed to correct flow rates with the net effect that is a redistribution that would minimise the divergence. If a large redistribution is necessary to achieve parity, then an incident may have occurred on the plant.

The emphasis in the design of the algorithms is on the development of a practical system, one that could easily be adapted for use on a real plant. A number of different activities were needed to convert the conceptual design into a practical additional safeguards system

A considerable amount of work has been spent designing and testing on real data virtually identical algorithms. This activity is not central to the work described in this thesis, and has been relegated to Appendix 1. However, it is evidence of the credibility of the algorithms on their ability to work in a real situation, and cannot be stressed too much.

In the absence of real plant data, the system has been tested using data from a hypothetical chemical separation area. These studies have demonstrated that the additional safeguards system described has the potential to be a valuable aid for inspectors responsible for nuclear materials safeguards.

Although unlikely to be implemented in its current form because of the lack of testing of the system on real data, the development of the system and the design of the tools presented in this thesis provide a set of guidelines for future developers. The importance of the data collection system has been underlined. As the foundations of any safeguards system, the collection of appropriate data at a satisfactory rate is crucial to the entire procedure.

The tools themselves are best thought of as examples of what would be required and certain of their component algorithms have previously been employed in analysing real plant data. The method developed for estimating gross multiplicative biases should provide a useful grounding for future development; again the development of the algorithm has been curtailed by the lack of real data.

ACKNOWLEDGEMENTS

The work presented in this thesis has progressed over three years with contributions from many individuals.

I would like to express my great appreciation to my supervisor, Dr John Howell, who has continually provided advice, guidance and encouragement during the course of this research. I would also like to thank Dr Stephen Scothern (formerly of the Department of Mechanical Engineering, University of Glasgow) for his advice in the early stages of the research.

I am grateful to many members of the International Atomic Energy Agency for their helpful insight during the development of the system specification, in particular, Dr R. Abedin-Zadeh and Ms T. Renis for useful discussions during the course of the work, Dr M. Ehinger for providing advice about plant operations, and Mr J. Plumb for supplying various sets of real plant data.

Many thanks also go to the UK R&D Support Programme to the IAEA, for supporting the work and to the United Kingdom Atomic Energy Agency, in particular to Dr Maurice Ward who manages this programme.

I would also like to thank the IT staff of the Department of Mechanical Engineering for all of the help and assistance they have provided during my time at the University of Glasgow.

Special thanks to Dr Jane Gardiner, my parents, and other members of my family, for the endless support, motivation and encouragement they have given me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.0 Overview

The operation of high throughput commercial nuclear reprocessing facilities has led to well-documented problems in the safeguarding of fissile material in the chemical separation areas (LASCAR 1992). In the case of the Rokkasho Reprocessing Plant (RRP), Burr & Wangen (1996b) have demonstrated that the annual material balance standard deviation based upon traditional monthly accounting will be of too great a magnitude to satisfy the protracted loss detection requirement specified by the International Atomic Energy Agency (IAEA). Thus there is a need for additional systems that enhance conventional accountancy and containment and surveillance systems.

One possible additional system has been proposed by Howell and Miller, (2001a), which is provided as an addendum to this thesis. The system performs something akin to dynamic fissile material accountancy, utilising dynamic mass balances to represent the progression of plutonium through the plant. This facilitates the detection of diversions of fissile material, providing additional assurances. One benefit of such a system is that it provides indirect verification of the conventional safeguards system.

At this juncture it should be stressed that the system is not intended to be accurate, being designed to have a resolution of the order of a kilogram rather than a gram of fissile material. The reason for this is twofold: firstly it would be very expensive to develop such a precise system, and secondly the false alarm rate has to be low to promote confidence in the performance of the system.

This thesis describes the underlying philosophy of this system and focuses on the development of several of its component tools.

1.1 Need for the System

1.1.1 Nuclear Safeguards

The International Atomic Energy Agency, one of the world's nuclear watchdogs, is responsible for policing The Treaty on the Non-proliferation of Nuclear Weapons (IAEA, 1992). If fissile material is diverted from a peaceful nuclear program for weapons purposes, the IAEA is supposed to detect the diversion in time to permit an international response before the diverted material is used to manufacture a device.

Therefore a basic goal of the Agency and thus by definition of nuclear materials safeguards is to declare with high confidence that a significant quantity or more of plutonium has not been diverted and if any diversion is detected, it is done so within a short period of time. The exact definitions of the period of time and a significant quantity are determined by various government policies, (see, for example, Islam et al, 1993). In general, the period of time would mean the order of a week. A 'significant' quantity is the same amount of material irrespective of the capacity of the plant in question, and is an extremely small proportion of the inventory in a large facility like THORP (see, for example, The Health and Safety Executive, 1995). For the Rokkasho Reprocessing Plant, the proportion is approximately 0.1%.

1.1.2 Practical Considerations

In order for the International Atomic Energy Agency to comply with the safeguards requirements set out in the Treaty on the Non-proliferation of Nuclear Weapons, most countries in the world have agreed to permit inspections of their nuclear facilities. IAEA inspectors visit facilities to assure themselves (and in turn the Agency) that material has not been diverted.

Although the countries have agreed in principle, conflict can occur between the inspectors and the plant operators caused by this imposition. Operators, although also concerned about safeguards, have different motives from inspectors; namely the efficient and safe operation of the plant on a commercial basis.

Inspectors may have to deal with what could be best described as information poor plants: for reasons of confidentiality and of cost, plant operators may be unwilling to disclose certain plant data or plant design information. Existing plants are often lacking certain instrumentation that would be desirable in an ideal case. The integrating of additional instrumentation into planned facilities is also problematic due to funding etc. Thus any additional safeguards system must be designed to utilise a minimum of instruments. Where available other instruments may then provide supporting evidence that the plant is running as intended.

1.1.3 Safeguards Systems

Conventional safeguards are based on two separate procedures: material accountancy, and containment and surveillance. Containment and surveillance is suitable for 'closed' areas like ponds and stores but is not appropriate for high throughput facilities that have a large number of identifiable paths that cross its external boundary. Accountancy is more appropriate for these areas. Accountancy has evolved into two different techniques.

1.1.3.1 Conventional Accountancy

Conventional accountancy compares the throughput of the plant with the change in physical inventory of the plant at set time intervals (Tsutsumi et al, 1982). Note that the balance is in mass of plutonium and that it does not take into account the flows of material. Using the nomenclature that any symbol with a ^ is a measured variable whereas any symbol with a ~ is an estimated variable, the balance can be written as:

$$\widetilde{MUF}_i = (\sum \hat{I}_{KMP_i} - \sum \hat{O}_{KMP_i}) - (\hat{M}_i - \hat{M}_{i-1}) \qquad (1.1)$$

where: MUF_i - material unaccounted for at the end of time period i

$\sum I_{KMP_i}$ - total mass in during time period i

$\sum O_{KMP_i}$ - total mass out during time period i

M_i - physical inventory of plant at end of time period i

The total mass of plutonium transferring in and out of the plant is measured in accountancy vessels, which are therefore known as key measurement points (KMPs). This is done whilst the plant is in operation. For the physical inventory to be known accurately, all the material within the plant has to be flushed from unobservable process units to tanks, whose inventories can be measured accurately. Due to this inconvenience conventional accountancy is only performed infrequently. The emphasis is on very accurate but infrequent measurements. The inventory measurements are based on a knowledge of the volume or mass of the solution contained in the tank, combined with a knowledge of its plutonium gravimetric (gm/gm) or volumetric (gm/l) concentration. Accurate estimates of these concentrations can only be obtained by taking samples, which are carefully analysed in laboratories.

Burr & Wangen (1996b) have shown that this approach is inadequate to meet IAEA goals.

1.1.3.2 Near Real Time Accountancy

Near Real Time Accountancy (NRTA) is an evolution of conventional accountancy with the balance being closed every five to ten days (Lovett et al, 1982). Others refer to NRTA balances of a month (Burr & Wangen, 1996b). Again the balance is in mass of plutonium and it works on the overall account; but now it is impractical to flush the physical inventory to tanks whose inventory can be measured, so the physical inventory remains in-situ. Again concentrations are derived from samples and it is this that largely limits the frequency at which the account can be closed. Because some of the inventory is impossible to measure, assumptions have to be made about the unmeasured inventories and the system must be designed to avoid a rise in false alarm rates. The MUF equation is then

$$\tilde{MUF}_i = (\sum \hat{I}_{KMP_i} - \sum \hat{O}_{KMP_i}) - \sum_{j=1}^{n} \Delta\overline{M}_j \qquad (1.2)$$

where: MUF - material unaccounted for at the end of time period i

$\sum I_{KMP}$ - total mass in during time period i

$\sum O_{KMP}$ - total mass out during time period i

$$\sum_{j=1}^{n} \Delta\overline{M}_j = \Delta\hat{M}_1 + \Delta\hat{M}_2 + \Delta\hat{M}_3 + \Delta\tilde{M}_4 + \Delta\tilde{M}_5 \ldots\ldots$$

$\Delta\overline{M}_j$ - change in physical inventory of unit j

(plant contains n units)

NRTA sacrifices accurate measurement of the physical inventory of the plant for an increase in the frequency of balance closure and hence in the timeliness to detect and resolve issues. Thus sensitivity is lost at the expense of the provision of a capability to detect and resolve an issue soon after it has arisen. Unfortunately, being based on a single MUF statistic, NRTA capabilities are limited to detection, these is no provision to diagnose. This is its major limitation.

1.1.4 Dynamic Accountancy

If it was to exist, dynamic accountancy might be considered to be the final evolution of the conventional accountancy approach. Put simply it is the continuous form of NRTA:

$$M\tilde{U}F(t) = M\tilde{U}F(0) + \left(\int_0^t \hat{I}_{KMP} - \int_0^t \hat{O}_{KMP}\right) - \sum_{j=1}^{n} \Delta\tilde{M}_j \qquad (1.3)$$

where: MUF - material unaccounted for

I_{KMP} - flow of mass through inlet

O_{KMP} - flow of mass through outlet

$\Delta\tilde{M}_j$ - estimated total change in physical inventory of unit over the time period

Note that $\Delta\tilde{M}_j$ would have to be used instead of $\Delta\hat{M}_j$ because, as will be discussed in the next sub-section, tank inventories are rarely measured directly because of the difficulties of measuring process concentration in-process. Estimation would then have to be based on an understanding of the flow of nuclear material through the plant. This would require an understanding of what actually constitutes a reprocessing plant, so to go any further it is important that the reader has a reasonable understanding of those aspects that are important here. Sections 1.2 and 1.3 are therefore provided for this purpose. The concept used here is that of inverse modelling, an introduction to which is given in section 1.4. The thesis returns to the main story in Section 1.5.

1.2 Plant Description

A nuclear reprocessing facility has five main areas: fuel disassembly and dissolution, chemical separation, product storage, conversion and waste handling. The work that is described here largely pertains to the chemical separation area although related work on the product storage area is described in Appendices 1 & 2.

The chemical separation part of a nuclear fuel reprocessing facility is normally operated continuously. Due to the lack of suitable on-line instrumentation, the separation of plutonium nitrate from other chemicals obtained when spent fuel rods are dissolved in nitric acid is controlled manually. To achieve this the various process stages are separated by sets of tanks consisting of a receiving tank (if fed from a continuous process), one or more buffer tanks and a feeding tank (if feeding a continuous process). Most of the inventory of the plant resides in these tanks and samples are taken from the buffer tanks for purposes of manual control. This results in an arrangement like that illustrated in diagram 1.2.1.

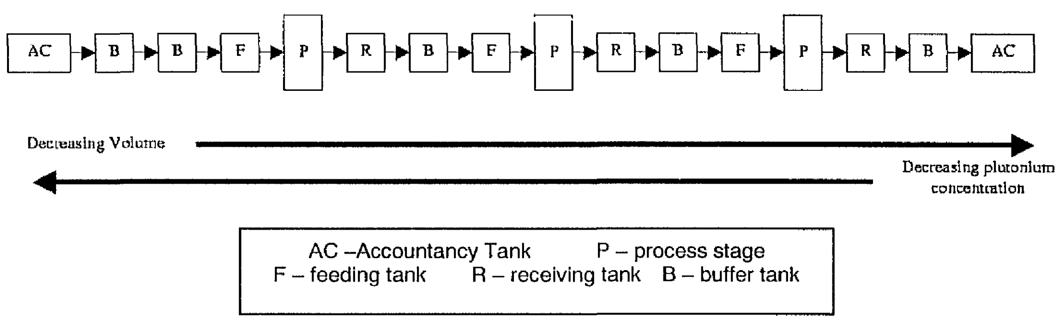


Figure 1.2.1 Arrangement of chemical separation area of reprocessing plant.

There are two main types of process stage: the solvent extraction plant (commonly known as a solex plant) and the concentrator plant. In terms of plutonium flow, the process stage transforms the plutonium concentration, [Pu], upwards with a corresponding reduction in flow rate, f, so that [Pu]f is maintained. Each plant component will now be discussed separately.

1.2.1 Tanks

1.2.1.1 Buffer Tanks

Buffer tanks are characterised by their short duration import and export flows. This results in their square wave volume profile (figure 1.2.1.1). Buffer tanks are usually situated between receiving and feeding tanks or at the head or tail of the plant.

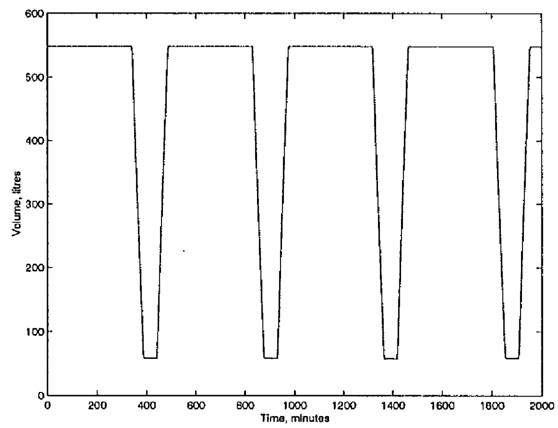


Figure 1.2.1.1 Buffer tank volume profile

1.2.1.2 Feeding Tanks

Feeding tanks are characterised by their short duration import and continuous (although not necessarily constant) export flows. This results in their reversed saw-tooth wave volume profile (figure 1.2.1.2). Feeding tanks are usually situated before any process stage that requires a continuous feed.

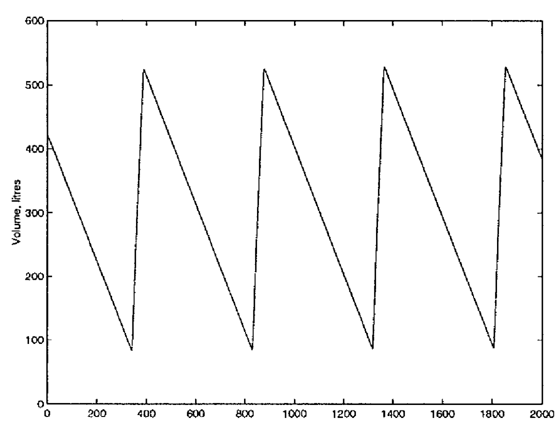


Figure 1.2.1.2 Feeding tank volume profile

1.2.1.3 Receiving Tanks

Receiving tanks are characterised by their continuous (although not necessarily constant) import and short duration export flows. This results in their saw-tooth wave volume profile (figure 1.2.1.3). Receiving tanks are situated after any process stage that has a continuous flow out.

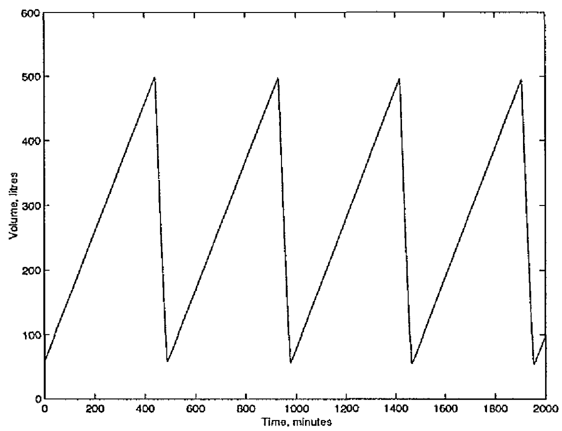


Figure 1.2.1.3 Receiving tank volume profile

1.2.1.4 Accountancy Tanks

Accountancy tanks at the boundaries are also known as key measurement points. Accountancy tanks are a variant of the buffer tank. The main difference is that the accountancy tank has additional and more accurate instrumentation, and means of homogenisation. Samples taken from the tank are analysed more closely. This means that any measured variable (including flow rates) are usually known to a higher degree of accuracy.

1.2.2 Solvent Extraction Plant

A solvent extraction plant consists of two or more cycles. In the first cycle plutonium and uranium is extracted from highly active waste, the plutonium is then separated from the uranium. Solvent extraction plants are operated in a continuous manner. Further cycles then improve the purity of the recovered plutonium.

A solvent extraction cycle consists of a number of pulsed columns. A technical description is given in Benedict, Pigford and Levi (1981). Each column has a separate function, e.g. oxidation, extraction, stripping but the underlying process is similar. The plutonium nitrate liquor is brought into contact in counterflow with an organic solvent to which the plutonium is preferentially transferred, leaving other components behind. By altering the valency of the plutonium so that it becomes insoluble in the organic solvent, the plutonium is transferred back into nitric acid in a separate pulse column. The majority of the plutonium migrates into the solvent or acid. The final pulse column utilises nitric acid; thus the exported liquor is plutonium nitrate.

The quantity of plutonium stored in a cycle at any instance in time cannot be measured directly but can be estimated on the basis of the known compositions of the feeds and on assumptions about the locations of the 'heavy metal fronts' in the columns. A certain level of confidence can be obtained about the position of these fronts by the use of neutron detectors, XRF detectors and density instruments. The liquor output from each stage of the solex plant (called a 'cycle') is largely composed of plutonium nitrate dissolved in nitric acid and at temperature T is of density ρ_s (Howell and Scothern, 1995):

$$\rho_s(T) = \rho_w(T) + \alpha_{Pu}(T)[Pu] + \alpha_{H+}(T)[H+] + ... \tag{1.4}$$

where: $\rho_w(T)$ - density of water (g/l)

$[Pu]$ - Pu concentration (g/l)

$[H+]$ - Acid molarity (mol/l)

$\alpha_{Pu}(T)$ & $\alpha_{H+}(T)$ - coefficients

The acid molarity affects the density and hence if the density measurements are to be of any use in the estimation of plutonium, molarity has also to be known. The acid molarity leaving a solvent extraction plant can be altered by adjusting the molarity of the final nitric acid feed. Knowledge of this molarity has to be obtained from the downstream receiving tank's density measurements.

1.2.3 Concentrator Plant

The concentrator plant consists of a storage tank, linked to a heating element via a downpipe and a riser. An outlet valve is fitted to the riser. Again a more technical description can be found in Benedict, Pigford, and Levi (1981). Liquor is imported into the storage tank from which material is passed through the heating element and returns to the tank. The heating action evaporates mainly water from the solvent and thus concentrates the material. Once the desired concentration has been reached, the material is exported through the outlet valve.

The factor of concentration is given by the ratio of import and export flows. The concentrator plant achieves significantly higher factors of concentration than a solex. Concentrator plants are normally operated in a continuous manner.

1.2.4 Sparging

To ensure that a homogeneous solution exists within a tank, nitrogen is sometimes bubbled through the mixture at regular intervals to agitate the contents. This also ensures that the liquor remains fully nitrated.

1.2.5 Sampling

Sampling is undertaken on buffer tanks, normally before the export of solution, to check the composition of the contents before it enters a process stage. If necessary the operator then corrects its chemical composition by addition of chemicals prior to export. Little of this sampling data is made available to the inspector, and when it is, a considerable time lag exists. This lag is longer for accountancy tank samples as these are analysed more closely. Material is removed from the tank, a small proportion taken for testing and the vast majority returned. This usually occurs one hour before export.

1.3 Instrumentation

There are various types of instruments available for measuring various parameters on the plant. These will be described by function.

1.3.1 Volume Measurement

Data relating to the status of the volume of a tank can be derived from a variety of sensors (BNFL 1996 and IAEA 1999a). The most common method is to use level dip-tubes.

An open-ended pipe is suspended in a tank and air or other gas is passed through the pipe. The pressure that is required to discharge liquid from this pipe is proportional to the head of liquid above the open-ended pipe and the density of the liquid. This pressure is measured by an electro-manometer and the change in the signal from the electro-manometer is proportional to the level within the tank.

With only a level dip-tube installed then various assumptions about the concentrations of the components of the liquor have to be made so that the volume, density and thus mass of the tank can be calculated.

1.3.2 Density Measurement

To avoid having to make these assumptions, a density dip-tube is installed in parallel with the level dip-tube (BNFL 1996, IAEA 1999a and Howell and Scothern 1995). The density dip-tube terminates at a known vertical separation above the level dip-tube. Its manner of operation is the same as the level dip-tube.

The density of the liquor present can be derived from the pressure differential between the two dip-tubes and the vertical separation.

1.3.3 Flow Measurement and other Instrumentation

Numerous other instruments might be available such as flow rate monitors (IAEA 1999a). Although the technology is established, operators avoid invasive monitoring systems due to the problems of having maintainable items in cells (expensive) and invading the process liquor (plant containment). Thus flow rate is usually only measured in inactive lines.

Flow meters are used within plants. The material entering a plant is metered (as opposed to measured). An example of this is a wheel on which measuring spoons are mounted.

1.4 Inverse Modelling

Inverse modelling has been defined as (Hess et al 1991):

Computational methods that determine control inputs to a dynamic system that produce desired system outputs.

Inverse modelling has long held the attention of a wide spectrum of scientists, from ecologists (Patten 1970) to aeronautical dynamists (Thomson and Bradley 1990). Inverse solutions are notorious for being non-unique and unstable and thus can be classified as ill posed. Inverse solutions are also influenced by any measurement biases present in the system. These problems are illustrated by example in the following sub-sections.

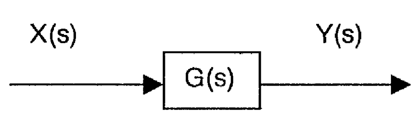
1.4.1 Non-uniqueness of Inverse Solutions

Inverse solutions have a propensity to be non-unique because it is often the case that the number of measured state variables is less than the number of inputs i.e. the systems are often under-specified. This results in non-unique solutions.

For example, consider a tank whose volume profile shows an increase of 1 litres/s. One possible inverse solution is that the flow in is 1 litre/s and the flow out 0 litre/s. However, this is not the only solution, the flow in could just as easily be 1001 litre/s and the flow out 1000 litre/s. The rate of change of the volume is only indicative of the difference in magnitude between the flow in and flow out.

1.4.2 Instability of Inverse Solutions

To illustrate the instability of an inverse solution, consider the Laplace domain system below:



where: $$G(s) = {7154(s^2 + s)} \Big/ {(s^4 + 178s^3 + 8027s^2 + 22850s + 15000)}$$

$$X(s) = 1/s \text{ (unit step at time = zero)}$$

This can be rearranged to give (see figure 1.4.1 for a plot of this function):

$$Y(s) = \frac{1}{(s+2)} + \frac{2.92}{(s+100)} - \frac{3.92}{(s+75)}$$

Suppose the output of the system Y(s) was estimated to be $\frac{1}{(s+2)}$ then the inverse solution for X(s) could be approximated to:

$$X(s) = \frac{1}{7154} * (s + 175 + \frac{7500}{s})$$

The first term in X(s) is the second derivative of a unit step (or the derivative of infinity) which is not physically meaningful.

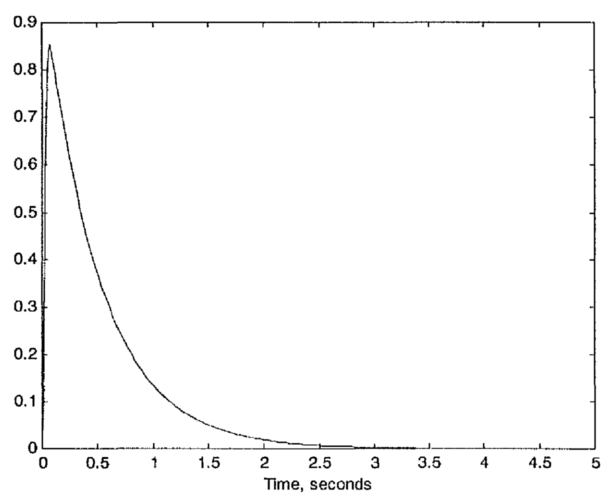


Figure 1.4.1 Plot of Y(s) when G(s) is subjected to a unit step at time zero

1.4.3 Affect of Measurement Bias on Inverse Solutions

Two possible types of bias may be present on the measured state variable used to estimate the system's inputs: additive and multiplicative. As far as tank volume measurement systems are concerned, additive biases are of little consequence as these do not affect the estimation of volume transfers. However, multiplicative biases can cause many problems if they are not identified and estimated, because the multiplicative bias will adversely affect the transfer estimates.

For example, consider a tank that has a constant volumetric flow in and an intermittent flow out. The volume of the tank is measured, but subjected to a multiplicative bias.

The inverse solution for either or both flows will be in error due to the bias on the volume measurement.

This problem is prevalent in other areas of research (Murray-Smith 2000).

1.4.4 Affect of Noise on Inverse Solutions

Inverse modelling is susceptible to noise. Consider G(s) in section 1.4.2, the inverse of which is s^4/s^2 which is susceptible to noise. Furthermore, the numerical differentiation of a measured state, which is subjected to noise, is extremely difficult and arduous. The possibility of producing a robust yet simple algorithm to perform such a task is low.

1.5 Proposal for an Additional Safeguards System

It should now be clear that the direct measurement of the plutonium inventory in a tank, or of its mass and plutonium concentration, is rare. Although mass can be measured readily, plutonium concentration cannot. Measurements can only be obtained via sampling, and this introduces significant overheads in terms of cost and laboratory time. The alternative is to estimate the plutonium concentration throughout the plant. The estimation of plutonium concentration requires an understanding of the movement of solutions through the plant, which in turn requires a model. Having constructed a model, it then seems quite natural to explore other possible uses because it seems pointless to distil all the knowledge now obtained into the single MUF statistic of Equation 1.3 i.e. the dynamic account. This leads to the following concept:

Having confidence in the accountancy tank measurements, the aim is to produce a system in which the differences between the variables associated with the modelled inventory and those associated with the measured physical inventory are justified. In cases where they are not justified, the system is intended to generate partial diagnoses to explain the disagreements observed. Thus the aim is to detect and diagnose rather than to improve the account.

The system proposed here can be viewed as an evolution of the solution monitoring systems previously developed at the University of Glasgow (Howell and Scothern 1997). In very broad terms, the approach is based upon simple plant simulations.

Simple simulations are used in a number of different roles: in generating reference data, in generating boundary conditions for other simulations, and in reasoning about disagreements.

The plant is then deemed to be operating as declared if a simulation can be constructed that is in agreement with the plant measurements over two different time intervals:

- Short-term: approximately equivalent to 1.5 cycles of filling/emptying a buffer tank.
- Medium-term: approximately equivalent to 10 cycles of filling/emptying a buffer tank.

Long-term assurances are not considered because other approaches may be more appropriate, e.g. conventional accountancy. Although the aim is not to improve the account *per se*, the methods presented in this thesis would improve the accuracy of the physical inventory estimates and hence of both the conventional and near-real time accounts.

An outline of the proposed system is given in Figure 1.5.1. As in Howell & Scothern (1997), the main output is that of a list of *events*, which describe those physical activities that would be needed so that the simulation correlates with the plant data. The key system components are a data collection system, a real-time database, a plant simulation and various tools that perform various functions: short-term assurances, medium-term assurances and redistribution. These various components are described in this sub-section.

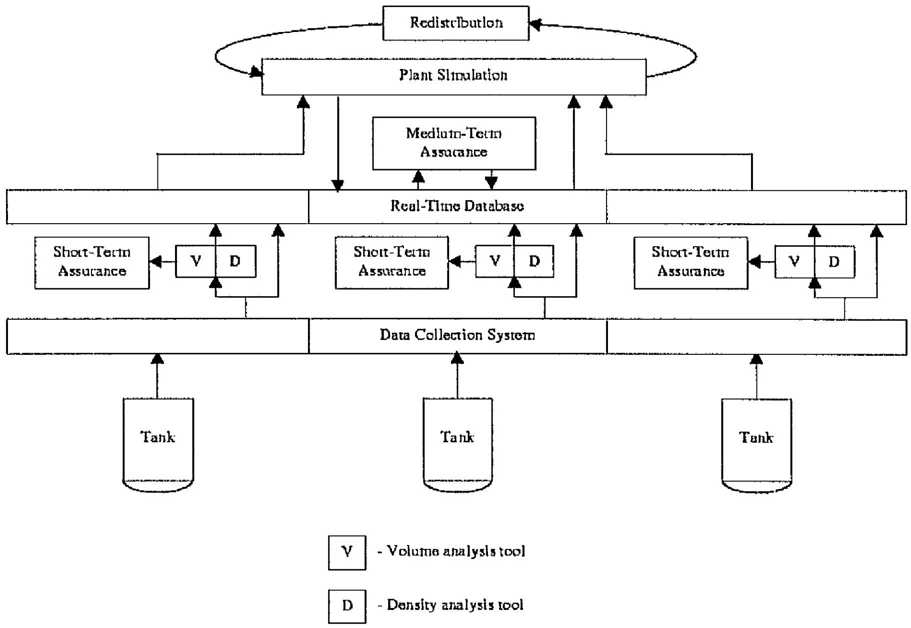


Figure 1.5.1 Outline of Additional Safeguards System

1.5.1 Definition of Terms

1.5.1.1 Event

As far as the system is concerned, an event is a set of one or more actions that are viewed by the operator or inspector as representing a single entity. It would not be unusual for the operator or inspector to attach a descriptive name to an event. For instance, the event 'transfer from Tank 1 to Tank 2' might be used to denote the opening of various valves, the starting of the transfer mechanism, eventually stopping the transfer mechanism and finally closing the valves. The observed 'symptoms' of the transfer are a decrease in the volume of Tank 1 and a corresponding increase in the volume of Tank 2. From a symptomatic point of view, an event can be viewed as a sequence of sub-events which can be described by a particular choice of diagnoses.

1.5.1.2 Sub-event

Refers to a single rise or fall in the measured value of a sensor attached to a plant element. The most common sub-events are increases and decreases in the volume of a tank.

1.5.2 Strategy

The base data obtained from the plant can be summarised as follows:

- accountancy tank data at inlet;
- volume, density and temperature measurements for each tank;
- accountancy tank data at outlet.

With no further analysis, our understanding of the plutonium distribution is simply the imported and exported quantities from the accountancy tanks, the region between them is simply a black hole into which material disappears and then reappears sometime later. Note also that only bulk volume and the combined density of all the solutes in the liquor are measured, not plutonium concentration.

1.5.2.1 Plant Simulations

The simulations largely derive from balances of water, plutonium, nitric acid and any other components where the choices of matter 'balanced' varies depending on the process units involved. There are two types of model, one that pertain to tanks and one that pertains to 'hidden' inventories like solvent-extraction cycles and the concentrator. The tank simulation is composed of a number of materials balances like:

$$\frac{dM_{Pu}}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu] \tag{1.5}$$

where f denotes a flowrate and [] a concentration. Based on the steady state follower approach, the hidden inventory models are very simple. These will be described in 5.1.

1.5.2.2 Estimation of Plutonium Distribution

Simulation is possible because material moves forwards from the input accountancy tank. The accurate knowledge of the plutonium exported from the input accountancy tank data will allow the simulation to propagate material through the plant. The simulation simulates each tank set and process stage in turns, enabling the system to estimate the plutonium distribution throughout.

The results of this simulation are then used to perform the short-term assurance simulation. Any short-term errors identified would then be rationalised.

In order to perform this simulation, the following information is required:

1. plutonium concentration from inlet;
2. import transfers from inlet;
3. import and export transfers from all tanks;
4. acid molarity exported by process stages;
5. addition of substances to tanks to alter liquor composition;
6. initial conditions of tanks;
7. initial load of process stages;
8. plutonium concentration from outlet;
9. export transfers from outlet.

The operator can supply 1, 2, 7, 8, and 9 whilst 6 is easily obtainable. New methods have had to be developed for points 3, 4 and 5. These are described in detail in Chapters Three, Four and Five, and the problems are outlined below. Point number 3 includes every import and export transfer in a tank, i.e. includes the unauthorised diversion of material.

1.5.2.3 Estimation of Transfers

Any information pertaining to the import and export flows of a tank must be obtained indirectly from analysing its volume measurement record. If a simple system model is used to represent this situation:

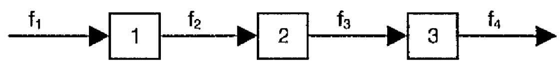
$$dV/dt = f_i - f_o \qquad (1.6)$$

where: V - output

f_i & f_o - inputs

it can be seen that the inputs to the system are to be determined from the system outputs – this is a definition of inverse modelling. For a single tank this is impossible as the volume only contains information pertaining to the difference between the import and export flows.

The connectivity of the plant together with the accountancy tank flows allows this problem to be solved. Consider the simple three-tank system shown in figure 1.5.1.1 below:



$$dV_1/dt = f_1 - f_2 \quad dV_2/dt = f_2 - f_3 \quad dV_3/dt = f_3 - f_4$$

Figure 1.5.1.1 Three-tank system

Without knowing any of the flows the problem is not solvable, there being three equations and four unknown parameters (f_1, f_2, f_3, f_4). The introduction of the accountancy flows (f_1, f_4) reduces the problem to three equations and two unknowns, which is now solvable.

The simplest method of inverse modelling for the liquor flow rates associated with a tank, would be to numerically differentiate the volume measurement record. Unfortunately, as this measurement is susceptible to noise, this approach would lead to instability. Therefore a more robust approach is required.

This approach is based on the assumption that during the horizontal sections of volume measurement data no flows are occurring. The inverse modelling task is simplified into two stages: the identification of the sections of volume measurement record that correspond to flows and the estimation of the flow rates of those stages.

1.5.2.4 Redistribution of Plutonium Inventory

For various reasons inverse modelling is not accurate (these were explored in section 1.4) and over a period of time any simulation that is driven by data derived from inverse modelling will diverge from reality. Again consider the three-tank system shown in figure 1.5.1.1. In this instance f_1 is a constant accountancy flow and the throughput of the plant is measured after tank 3. Estimates for the constant flow rates f_2, f_3, f_4 have been obtained by inverse modelling. If f_4 is over-estimated by 1%, the error in throughput between the real and simulated plant is that shown in figure 1.5.1.2.

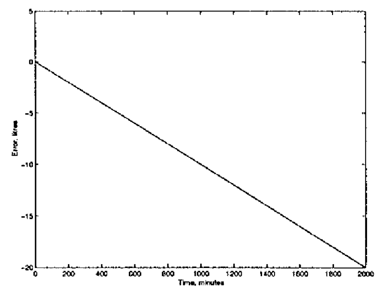


Figure 1.5.1.2 Divergence of simulated throughput from actual throughput

If the other flows (f_2, f_3) are also badly estimated, all of the tanks affected by the incorrect flow rates will diverge from their volume measurements. If f_4 is now replaced with the accountancy flow, the throughput of the simulated plant will match that of the actual plant. However, the individual tanks will still diverge from reality. By altering the estimated flow rates logically to redistribute the errors between the tanks, the simulation can be made to converge with the real plant.

This may require the introduction of additional events e.g. diversions of material.

1.5.2.5 Effect of Tank Biases

One reason for poor flow-rate estimation and hence model divergence, are multiplicative measurement biases. If these exist it is more sensible to estimate them and hence compensate than to redistribute. The decision as to when to redistribute and when to estimate biases is a complex one. It is preferable that biases are estimated as a redistribution of large amounts of material may result in false alarms.

1.5.3 Implementation

In section 1.5.1, the basic outline of the additional safeguards system was established. The tasks that the system has to undertake are:

- Basic Tasks
  - Estimate the import and export flow rates for each tank.
  - Estimate the acid concentration output from a solvent-extraction cycle.
  - Simulate each process unit.
- Short-term Assurance Tasks
  - Detect and diagnose disagreements.
- Medium-term Assurance Tasks
  - Redistribute plutonium.
  - Detect and diagnose disagreements.
- Additional Task
  - Estimation of multiplicative volume measurement biases.

To perform the first basic task, a tool, which analyses the volume measurement record of a tank, is required. Since the volume profiles of the feeding and receiving tanks are closely related, a single tool can be used for both. An additional tool is required for buffer tanks. The second basic tool analyses the density measurement record from receiving tanks.

The short-term and medium-term assurance tasks make use of the same family of tools. The detection and diagnostic tools make use of a variety of algorithms, e.g. cumulative sums based detectors, observer-based detectors, observer-based diagnostic tools. The redistribution tool is associated with the medium-term simulation and simply redistributes the small plutonium disagreements to improve the systems understanding of the plutonium distribution in the plant.

Finally, an additional tool is required to estimate the multiplicative biases that affect the volume measurements. Since transfers used in the simulations are derived from the volume measurements, any multiplicative bias that affects that measurement also affects the plutonium distribution. This results in a divergence between the model and actual plant in the medium-term.

All of the tools are designed to be modular in construction. The reasons for this are as follows:

1. Quality assurance of the software implementation.
2. The need for further development
    a). Quality of real plant data is unknown.
    b). Added complexity may not be of benefit.
    c). The flow estimation tools have common components.

1.6 Original Work

1.6.1 Safeguards

The development of additional nuclear materials safeguards tools is, by necessity, extremely practical, international diplomacy cannot be based solely on theoretical arguments. There have been at least four attempts to monitor the movements of solutions in reprocessing plants: Cobb et al (1981a, 1981b), Ehinger et al (1981 and 1989), Burr and Wangen (1996b), and Scothern & Howell (1997). Both the Cobb and Ehinger activities focused on the AGNS facility in South Carolina. Both Scothern and Howell (1997) and Burr and Wangen (1996b) worked on solution monitoring systems for product storage areas.

Cobb et al. effectively sought the formation of the dynamic account by recourse to advanced instrumentation, Kalman filtering and other statistical estimators. Ehinger worked on a more practical approach, selecting and developing specific applications of process monitoring with a role in international safeguards. Howell (Automatica, 1994) was the first to examine the possibility of applying computer simulations directly, whilst Burr & Wangen (1996a) looked at some theoretical implications. In particular the work of Scothern & Howell was simulation based. By their very nature product storage facilities are very different from chemical separation facilities; tanks in one are virtually always static, the contents of a large number of tanks in the other are continually changing. The development of the simulation-based safeguards approach described here is completely novel.

The emphasis of this thesis is on the development of a practical system, one that could easily be adapted for use on a real plant. A number of different activities were needed to convert the concept described in 1.5 into a practical additional safeguards system (Table 1.6.1)

Number	Activity
1.	Analysis of volume measurements using inverse modelling.
2.	Analysis of density measurements using inverse modelling.
3.	Short-term disagreement detection.
4.	Diagnosis of short-term disagreements.
5.	Medium-term disagreement detection.
6.	Diagnosis of medium-term disagreements.
7.	Establish a method of identifying gross biases in a dynamic environment.
8.	Test system on realistic data.
9.	Investigate data collection systems.

Table 1.6.1 Table of required activities

With practical implementation as its goal, the quality of the measurement records is of key importance. Towards this end a considerable amount of time was spent working with real data collected from a commercially operated facility (Appendix 1) whilst gaining an insight into the performance of the Howell & Scothern (1997) system. This work is described in 1.8 (activity 9).

1.6.2 Model-based Fault Detection

A considerable amount of literature has been published about model-based fault detection and diagnosis. Two approaches exist in model-based fault detection and diagnosis: quantitative and qualitative methods. A survey of not only these approaches but of process fault detection and diagnosis as a whole exists in Venkatasubramanian (2001). There also exist some hybrid methods that seek to combine the best features of both quantitative and qualitative to their advantage.

The quantitative approach is an established research topic and uses techniques from control theory and statistical analysis such as observers (Frank, 1990), Kalman filters (Willsky, 1976; Basseville, 1988), parity equations (Gertler and Singer, 1990; Gertler, 1991), and parameter estimation (Young, 1981; Isermann, 1984). The quantitative approach is essentially a two-stage process. The first stage generates residuals between the actual plant and the expected behaviour of an explicit model of the monitored plant. These residuals are 'pseudo-signals' reflecting the potential faults of the system. The second stage chooses a decision rule for diagnosis.

The drawbacks of the quantitative approach are the difficulties related to modelling the process and the specific modelling of faults. The additional safeguards system described in this thesis can be thought of as a physical quantitative model-based approach. Geiger et al (2001) describe another type of physical quantitative model-based approach, albeit for a combined gas/oil pipeline scheme. This uses neural nets to correct for modelling inaccuracies.

To overcome these drawbacks, qualitative modelling has been adopted in recent years (Frank 1994; Forbus, 1984). However, the qualitative modelling approach generally leads to multiple solutions, which are not compatible with diagnosis. To overcome the limitations of individual strategies, hybrid methods have been developed that integrate the best features of each method. An example of this is the method developed by Gentil and Montmain (2000) that relies on both a qualitative causal representation of the process function and quantitative local behavioural models. However, this method still relies upon the existence of a precise model of the industrial facility.

1.7 Outline of Thesis

In Chapter Two, a review of the algorithms like Shewhart Control Charts, Recursive Least Squares used to construct the tools is presented. The emphasis is on simple, adaptable algorithms.

The designs for the volume analysis tools for both the feeding/receiving tanks and buffer tanks are discussed in Chapter Three together with suitable short-term detection and diagnostic tools (activities 1, 3, & 4). Chapter Four discusses the design of the density measurement analysis tool for receiving tanks. Again short-term detection and diagnostic tools are presented (activities 2, 3 & 4).

The medium-term assurance tools are discussed in Chapter Five. These are the distribution simulation, the redistribution tool, and the detection and diagnostic algorithms (activities 5 & 6). Presented in the same chapter is a method of estimating gross measurement errors to improve the accuracy of the system (activity 7). In Chapter Six, test cases and their results are presented and discussed (activity 8). The conclusions and recommendations for further research are included in Chapter Seven.

1.8 Practical Issues

Since the focus is on practical implementations, this chapter finishes by examining some of the issues involved.

1.8.1 Data Acquisition System

The data acquisition system is one of, if not the, most important components of a safeguards system. The data acquisition system is the foundation of the additional safeguards system and its performance will have far reaching implications for the overall system. The system is composed of both hardware and data collection software.

The minimum data to be collected by the system includes pressure dip-tube measurement records pertaining to tank levels and densities plus temperatures (IAEA 1999b). The data has to be recorded at a rate sufficiently fast to enable appropriate data analyses to be made. It is important to have a high data collection rate (e.g. every 15 seconds) to view the start and finish of process operations clearly. However this results in large quantities of data, most of which is of no relevance to safeguards. Hence data compression is important. The data acquisition software then converts this data into volume/density/temperature records together with their standard deviations. For safeguards purposes it is essential that any data point recorded can be traced back to its origin. Thus 'corrected' data cannot be recorded, as their legitimacy is questionable.

Operators are reluctant to publish information pertaining to their systems in situ for security reasons. However, details of both the JA6 system installed in the product storage area of Tokai-Mura Reprocessing Plant (Howell and Miller, 2001b) and the portable Volume Long-Term Monitoring device (known as VLTM) developed at JRC Ispra (Landat et al, 1997a, 1997b, and 1998) are available. Data collected on the TRP facility using both systems was made available for the work described here.

1.8.2 Ideal Data Acquisition System

Below is a list of features that would exist in the ideal data acquisition system for the additional safeguards system. If implemented, this data acquisition system would reduce the complexity of the overall system.

- Individual SEVA pressure transducers for each dip-tube pressure line.
- Software data compression only with user specified parameters.
- High data collection rate (e.g. every 15 seconds).
- Sparging interlock.
- Raw data storage for sensor validation/fault isolation
- Synchronised data collection for all tanks.
- Real-time data base for storage of measurements.

The reasoning behind this specification is outlined in the sections that follow.

1.8.3 Data Collection Hardware

Accurate pressure sensors are extremely expensive and because of this the JA6 system makes use of a type of rotating manifold known as a scanivalve. This mechanical device multiplexes pressure lines from various dip-tubes onto the same pressure transducer. Combined with the data collection software, the resulting system produces poor quality data that requires substantial pre-processing before any subsequent analyses can be made. The main problems are a loss of sensitivity, a loss of definition, a low rate of data collection, and a lack of synchronisation between tanks. For full details see Appendices 2 and 3. For these reasons, a system where a pressure transducer is used for each pressure line is preferred.

1.8.4 Data Collection Software

It is desirable that the data collection software includes data compression algorithms that are flexible in design. The need for data compression is not peculiar to solution monitoring. Data compression is used in many areas and in particular in the process industries (Hale and Sellars, 1981; Bristol, 1990; Cheung and Stephanopoulos, 1990a & 1990b). Data compression for safeguards purposes (Appendix 1) is peculiar because the objective is somewhat different: process industry data compression is about recording the under-lying trends in the data whereas, data compression for safeguards purposes should reflect more the quantitative state of the contents of the tank.

A number of user specified parameters, such as those available in the VLTM system (Table 1.8.4.1) are required to enable the system performance to be optimised. Appendix 4 shows the effect of varying a single, key parameter 'Alarm Threshold' on the quality of the data collected. Although not essential, it is useful if data collection was synchronised between tanks.

Parameter	Description
Acquisition time	Time to capture a single averaged measurement.
Alarm Threshold	Limit of change in measurement before it is recorded. A value of zero will ensure maximum data collection rate.
Backup time	If no measurement is output during this period, a measurement is recorded.
Maximum data collection rate	Acquisition time * number of multiplexed channels.

Table 1.8.4.1VLTM data acquisition system parameters

Also desirable is a sparging interlock. When a tank is sparged, there is an artificial rise in level (and thus volume) as the nitrogen bubbles through the liquor. By using an interlock it is possible to avoid recording this data, substituting previous data values for those that are affected by the sparging.

The data collection software must also be designed to accommodate eventualities such as the tank levels dropping below the density dip-tubes, which does not appear to be uncommon in normal operations.

The level/density measurement system has a tendency to be self-validating because the two key dip-tube pressure signals are highly correlated. To reduce the complexity of the system, and because it is sensible anyway, it is preferable that faults internal to dip-tube instrumentation be isolated separately. Adherence to the recently proposed standard on SEVA sensors (Clarke & Henry) is beneficial, as it would provide a communication protocol between the sensor validation system and the data collection software. Ideally the raw data would be recorded as well as the converted data. The raw data would be used to validate the operation of the dip-tube system.

1.8.5 Simulated Data

Due to the unavailability of real plant data pertaining to a solvent-extraction facility during the course of this research, appropriate time-series were obtained from a simulation of an actual facility. The simulation produces unbiased time series.

The data collection system was assumed to have an alarm threshold of 0.0, an acquisition time of 3 seconds and a maximum of 5 multiplexed channels. The volume, density and temperature of each tank were recorded into a database every 15 seconds. For the accountancy tanks, the accountancy flows as well as plutonium concentration was recorded in another database together with the loads of the solvent extraction cycles and concentrator. Noise and biases could subsequently be added to this raw data.

The simulation was developed to simulate various events that may take place on a real facility. An event is those instances that are deemed to be abnormal in some way. All events are of a temporal nature (i.e. have a start and stop time). For instance material may disappear at one time and re-appear at another. Events are classified by the time period over which they exist. Abrupt events have a relatively short duration whereas gradual events are long term trends.

CHAPTER 2

TOOL COMPONENTS

2.0 Introduction

In this chapter, various component algorithms are described which form the building blocks from which some of the tools are constructed. As the algorithms are common to more than one tool, it is important to introduce them at this stage. Knowledge of the algorithms is helpful for understanding Chapters Three to Five, where the tools are described in detail.

The algorithms are drawn from a number of fields, although the majority are widely used in statistical process control. None of the algorithms are unusual or novel in anyway. The emphasis is on practice rather than theory.

The algorithms described here have been chosen on the basis of our limited exposure to real data during the construction of a new and original data compression tool, albeit from a product storage area (Appendix 1). The function of the tool developed for that application was to extract the pertinent features from the measurement record whilst discarding the unimportant data. One aspect of this 'feature extraction' tool was the identification and recording of abrupt transfers, a not dissimilar task to one required by this system.

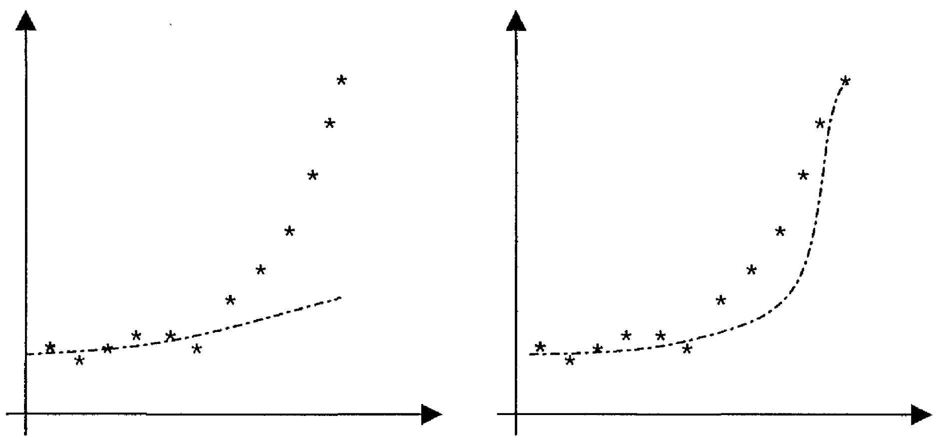
The algorithms are described in the context of their application to fulfil the tasks required by the system.

2.1 Estimation of Moving Average

As outlined in section 1.6, the system has to estimate the transfers into and out of a tank from the volume measurement records. To achieve this, the points at which transfers begin and end have to be identified.

The method used is to detect divergences between the plant data and an estimated process mean. This estimated process mean is the moving average of the plant data. If the measurement data diverges sufficiently then a change can be detected. If the noise on the real plant data is asymmetric then the moving average estimate will be biased. This will result in incorrect transfer estimates. Therefore, pre-filtering of the plant data is required in this situation. The simulated 'real' plant data used in the development of the system has a symmetrical noise profile, so pre-filtering is not an issue.

The choice of moving average estimator is dependent on two factors. Firstly, for reliable detection of changes, the plant data must diverge significantly. Therefore the estimator needs to be insensitive; i.e. the tracking of the data has to be poor. (Figure 2.1.1). The second is that this insensitivity must not impair the estimator's ability to track uphill or downhill segments of data as this will result in poor transfer identification.



----- : Moving Average * : Data Point

Figure 2.2.1 Poor tracking of the data during changes (left plot compared with right plot) results in reliable change detection.

The exponentially weighted moving average (EWMA) or geometric weighted moving average (GWMA) is used extensively in forecasting and time series modelling (Roberts 1959 and Crowder 1987). The EWMA is defined as:

$$z_i = \lambda x_i + (1 - \lambda) z_{i-1} \qquad (2.1)$$

where: z_i : exponentially weighted moving average.

x_i : data point

$0 < \lambda \leq 1$: a constant

However, to obtain the insensitivity required for reliable detection, the value of λ is small. This results in poor performance on uphill or downhill segments as the data increases or decreases more rapidly than the EWMA.

As the EWMA is unsuitable, a minimum-variance unbiased estimator is preferred. Ignoring process noise, a volume measurement history can be represented by a series of straight-line segments, the interception points of which correspond to physical events on the tank. An ideal estimator to generate equations for these straight-line is the recursive least squares (RLS) or sequential least squares (SLS) algorithm (Graup 1972 and Hisa 1977). The algorithm has the desired insensitivity and uphill/downhill tracking ability. The RLS estimate requires to be reset when a change is detected. This concept has already been shown to work in Appendix 1. A disadvantage of this method is that the fitting of straight lines to data that may taper asymptotically may cause problems.

2.1.1 Recursive Least Squares

The recursive least squares algorithm estimates the gradient, m, and constant, c of a straight line [i.e. m,c : $v_n = mt_n + c$] . Graup defines the algorithm as follows:

$$x_{n+1} = x_n + P_{n+1}h_{n+1}(v_{n+1} - h_{n+1}^T x_n) \tag{2.2}$$

where:

$$v = h^T x + n$$

$$h = \begin{bmatrix} t & 1 \end{bmatrix}^T$$

$$x = \begin{bmatrix} m & c \end{bmatrix}^T$$

n - noise

$$P_{n+1} = P_n - P_n h_{n+1}(1 + h_{n+1}^T P_n h_{n+1})^{-1} P_n h_{n+1}^T$$

To apply this, $P_0 = \infty$ * 2x2 identity matrix and the elements of x_0 are m = 0 and c = the first data value. To reset the RLS calculation, $P_{n-1} = P_0$ and x_n to have m = 0 and c = value of process mean at this point. Resetting the RLS calculation is preferred to the introduction of a forgetting factor (Soderstrom and Stoica, 1989) as this introduces similar sensitivity issues to the EWMA.

Using delayed values from several points beforehand can increase the insensitivity of the RLS estimate. This may be of use when the data collection rate is high in comparison with the plant dynamics.

2.2 Detection of Changes

Detectors are distributed throughout the system. For example to detect plant data diverging from the process mean estimate or an alarm signal being activated. The changes that are of interest to the system can all be classed as abrupt.

The classic abrupt detectors are the Shewhart Control Chart (Shewhart 1931), the Cumulative Sum Control Chart (Page 1954) and the V-mask (Barnard 1959).

2.2.1 Shewhart Control Chart

The Shewhart control chart (Shewhart 1931, Montgomery 1996) is an established method for online process monitoring. If the process data values fall within the upper and lower control limits (UCL and LCL) then the process is deemed to be in control (or not changing). If the data values move outside the control limits it is an indication that the process has moved off target. The Shewhart control chart is defined as:

$$\begin{aligned} UCL &= \bar{v} + L\sigma_v \\ LCL &= \bar{v} - L\sigma_v \end{aligned} \qquad (2.3)$$

where: v - data value

$\bar{v}$ - mean of v

σ_v - the standard deviation of v

L - 'distance' of control limits from $\bar{v}$

A major disadvantage of the Shewhart control chart is that it only uses the information about the process contained in the last data point and it ignores any information given by the entire sequence of points.

To overcome this disadvantage whilst maintaining the simplicity and ease of understanding, the Shewhart control chart can be combined with that of RLS estimation, as proposed in Basseville & Nikiforov (1993).

2.2.2 Standardised Cumulative Sum

The standardised cumulative sum or cusum (Page 1954, Chatfield 1996) is an established method for online process monitoring. Cusums combine successive measurements, which enables them to detect small changes in the process.

The standardised cusum consists of two cusums, the one-sided upper (C^+) and lower (C^-) cusums. The standardised cusum is calculated as follows:

$$\begin{aligned} C_n^+ &= \max[0, y_n - k + C_{n-1}^+] \\ C_n^- &= \max[0, -k - y_n + C_{n-1}^-] \end{aligned} \tag{2.4}$$

where: $y_n = \dfrac{v_n - \bar{v}_n}{\sigma_v}$

v - data value

$\bar{v}$ - mean of v

σ_v - the standard deviation of v

k - reference value

If either cusum exceeds the decision interval h, the process is deemed to be out of control. The choice of k and h affect the performance of the cusum.

2.2.3 V-Mask

The V-mask (Barnard 1959 and Johnson 1961) is an adaptation of the standardised cusum. Using the v-mask simplifies the procedure for choosing k and h for the standardised cusum. The v-mask is applied to successive values of the upper and lower cusums (C^+ and C^-), as illustrated in figure 2.2.3.1. The V-mask is placed on the plot with A on the last point of C^+ and AB parallel to the t axis. If all the previous values for C^+ lie within the two arms, the process is in control. The process is deemed to be out of control if any of the cumulative sums lie outside the arms of the V-mask.

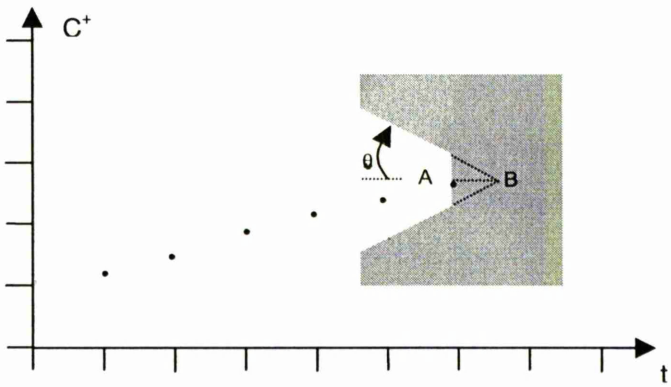


Figure 2.2.3.1 example of V-mask

Barnard and Johnson both suggest techniques for tuning the V-mask, but here the procedure described by Montgomery (1996) is preferred. No matter which technique is used, some trial and error is necessary until the desired performance is achieved.

Let AB = d, the lead distance; the performance of the V-mask is determined by d and θ, the angle of the arms to the horizontal. The values of k and h are related to these values by the formulae:

$$k = C \tan\theta$$
$$h = dk$$

C is the horizontal distance on the V-mask plot between successive points in terms of unit distance on the vertical scale. By selecting a constant d, varying θ will affect the performance of the V-mask.

2.3 Calculation of Flow Rates

Once a transfer has been identified, the flow rate during the transfer has to be estimated. Note that this need not be performed in real-time so both optimisation and estimators are considered. The choice of algorithm is influenced by the quality of the measurement data. If the data is relatively clean or has been pre-filtered for the reasons outlined in section 2.1, then a simple averaging algorithm is sufficient. Otherwise, a more complex algorithm is required.

2.3.1 Average Flow Rates

Although obvious, the averaging method of flow rate calculation is presented for completeness. If the volumes at two time points are known, then the average flow rate can be calculated:

$$\bar{f} = \frac{v_2 - v_1}{t_2 - t_1} \tag{2.5}$$

where:

$\bar{f}$ - averaged flow rate

t_1 - start time of flow

v_1 - volume at start of flow

t_2 - end time of flow

v_2 - volume at end of flow

2.3.2 Function Optimisation

The calculation of flow rates can be defined as a function optimisation problem. Optimisation problems consist of two components:

- the function to be optimised which has to have either a maximum or a minimum at the desired values;
- the optimisation algorithm.

Several numerical methods for optimisation exist; three possibilities are Powell's method, the Downhill Simplex method, and simulated annealing.

For a set of given parameters, the function to be optimised simulates the volume of a tank, returning the cumulative sum of the square of the errors between the simulated volume and actual volume of the tank i.e.

$$E_n = E_{n-1} + (Vs_n - Vm_n)^2 \qquad (2.6)$$

where: E -cumulative sum of errors squared
Vs - simulated volume measurements
Vm - actual volume measurements

If the simulated volume exactly matches the actual volume measurements, then the function will be at its minimum value - zero.

The more popular optimisation algorithms such as the Downhill Simplex Method (Dantzig 1963 and Nelder & Mead 1965) or direction-set methods, of which Powell's method (Powell 1964 & 1968) is the well-known prototype, are not able to differentiate between a local and the global minimum. Therefore the optimisation algorithm utilised here is the simulated annealing algorithm (Van Laarhoven 1988 and Van Ginneken and Otten 1989) which statistically guarantees an optimal solution (i.e. the global minimum is always identified).

Simulated annealing is a method of finding optimum solutions to problems that have a large set of solutions, in an analogous fashion to the physical annealing of solids to attain minimum energy states. The fundamental idea is to generate a path through the solution space, from one solution to another nearby solution, leading ultimately to the optimum solution. In generating this path, solutions are chosen from the locality of the preceding solution by a probabilistic function of the improvement gained by this move. A key element of the simulated annealing algorithm is the Metropolis function (Fishman 1996 and Barkema & Newman 1999).

2.4 State Space Based Methods

Both Kalman filters (Kalman 1960, Kalman and Bucy 1961) and observers (Luenberger 1966 and 1977) are dynamic systems whose state variables are the estimates of the state of the target system where the Kalman filter can be considered to be an optimal observer.

Although Kalman filters can be constructed to undertake input estimation for a tank system, the resultant estimate would be inseparable, i.e. the rate of change of the state is estimated so that the sum of the flow in and flow out would be estimated. The additional effort required to separate the flows and to construct the Kalman filter in the first place mean that they are only likely to be used in situations if excessive noise is present. Since real data is not available they were not considered any further. There may also be problems with computational time and the numerical stability of the solution (Chen, 2000).

Although the plant data would be discrete, for convenience, any simulated test data was converted to a continuous form via linear interpolation for the analysis. A discrete analysis could have been performed, but linear interpolation would still have been necessary to ensure a reasonably short interval between solutions.

Observers are used in several different guises in the system tools. The standard state-space equation (Friedland 1987) for an observer is:

$$d\hat{x}/dt = A\hat{x} + Bu + K(y - C\hat{x}) \qquad (2.7)$$

Observers are widely used in all aspects of control systems. Throughout the system they are used as:

- Estimators
- Components of detectors

The classic application of an observer is as an estimator, so this will not be elaborated upon as the majority of control textbooks cover this topic (Friedland 1987, Rugh 1996). The incorporation of observers into detectors is less well known and so is described in some detail.

The approach described here is a variation of fault detection using observers (Patton et al 1989, Viswanadham and Srichander 1987, and Viswanadham and Minto 1988). In general, these observer schemes are designed for sensor, actuator and component fault detection and isolation in dynamic systems. The important aspect of this work is the generation of so-called residuals, i.e. of functions that carry information about the faults, and then analysing the residual to see if a fault has occurred.

These philosophies can be modified for use in the safeguards system. The basic idea is to generate a residual that remains at a nominal value when the plant is operating as expected.

If the residual diverges from its nominal value, then decisions can be made about the plant's operation. For example, consider a simple tank:

$$dV/dt = f_i - f_o - f \qquad (2.8)$$

where: V - volume of tank

f_i - measured flow into tank through normal inlet

f_o - measured flow out of tank through normal outlet

f -unmeasured flow through a separate outlet

The observer equation becomes:

$$d\tilde{V}/dt = \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} f_i \\ f_o \end{bmatrix} + K(V - \tilde{V}) \qquad (2.8)$$

where: $\tilde{V}$ -observer estimate of volume

K - observer gain.

which can be represented as a block diagram (figure 2.4.0.1). If, in normal operation, the unmonitored outlet is never used, then the residual ($K(V - \tilde{V})$) is approximately zero. However, if material is removed via the unmonitored outlet, the residual becomes non-zero as the observer compensates for the unknown flow. This change in the residual can then be detected.

The choice of K is dependent upon the performance criteria of the observer and the quality of the real data. Therefore the decision of the value of K is left to the implementers of the system.

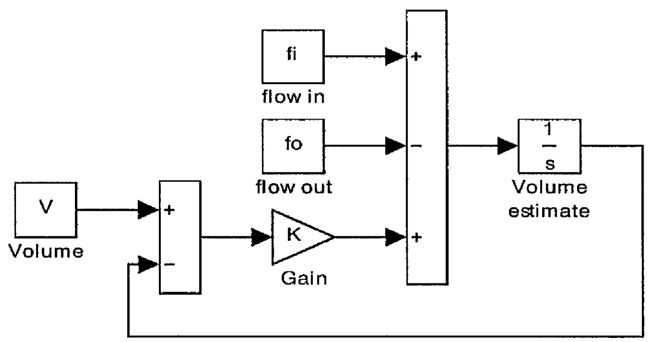


Figure 2.4.0.1 Block diagram of tank observer

As plant dynamics are involved, it is unlikely that discontinuities will exist in the plant data so K can be of a relatively low value providing the tracking ability of the observer is not impaired.

In most instances (like that described above), the observer wiii the noise on the measurement signal during normal operation. To select a suitable value for K, a section of noisy data can be input into the observer and K adjusted until the error is below an arbitrary value.

2.5 Identification of Points of Change

Since the detectors only detect after the change has occurred, some method of identifying the data point closest to the actual point of change is required. Three possible methods for doing so are presented here.

2.5.1 Interception Point of Straight Line Segments

If the volume history can be represented by a series of straight-line segments, then an estimate for the points of change will be given by the intercepts of these lines. Using the values of m and c from the RLS algorithm, the intercept time is given by:

$$t_i = \frac{c_2 - c_1}{m_1 - m_2} \tag{2.9}$$

where: t_i - intercept time

m_1, c_1 - coefficients of line 1

m_2, c_2 - coefficients of line 2

To avoid the 'real data' problem (section 1.8),the data point with the time nearest to t_i should then be selected. A disadvantage of this method is that the estimates for m and c will not be immediately available after a point of change has been detected, thus the identification of points will lag one point of change behind the detector.

2.5.2 Simulated Annealing Parameter

The function to be optimised by simulated annealing can be written so that an input parameter could be the time at which a change in flow rate occurs. Thus the optimised solution would contain an estimate for the time of change. Again the data point nearest to this time should be selected.

2.5.3 Rectangle Search

Although outwardly simple, it appears that the algorithm proposed here is unpublished. The rectangle search is only suitable for buffer tanks where the volume is constant before and after a transfer. The point at which the change is detected is shown as point E in (figure 2.5.3.1). A rectangle (ABCD) is constructed around point E with two of its vertices being data points Δt from E (A and C in the figure).

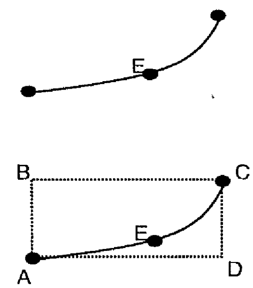


Figure 2.5.3.1 Illustration of rectangle search

If the data points are at A and C, the transfer is into the tank. If BE > ED, then point A represents the start of the transfer. Otherwise point C represents the end of the transfer.

Similarly, if the data points are at B and D, the transfer is out of the tank. If CE > AE then point D represents the end of the transfer. Otherwise point B represents the start of the transfer.

CHAPTER 3

VOLUME MEASUREMENT ANALYSIS

3.0 Introduction

In this chapter, the tools developed to analyse the volume measurements from buffer and feeding/receiving tanks are described. The tools share a common kernel of components, only when calculating the flow rates do they significantly differ. As the analysis of the flow rates is performed on a per cycle basis (Section 1.5), the opportunity is taken to detect short-term anomalies. The three sub-sections of this chapter are:

1. calculation of flow rates;
2. detection of short-term disagreements;
3. diagnosis of short-term disagreements.

A separate tool was developed to extract the volumetric flow rates from buffer tanks and feeding/receiving tanks. The detection and diagnostic tools are similar for both types of tank.

With reference to section 1.6, this chapter contributes activity 1 in its entirety, the analysis of volume measurements using inverse modelling. It also contributes towards activities 3 and 4, the short-term disagreement detection and diagnosis activities. Of these contributions, all are original.

3.1 Analysis Strategy

With no diversions taking place, two types of error can exist when simulating a tank using inverse modelled flow rates:

- Transient: where the total volume of liquor transferred is correct, but the duration of the flow is in error.
- Residual: where the total volume of liquor transferred is in error.

Obviously, residual errors must be minimised otherwise the simulated plant will diverge from the real plant in a short space of time. However, the complete elimination of residual errors is not possible, hence the need for the redistribution tool. (Section 1.5.2.4).

Residual errors exist because of inaccuracies in the estimation of the total volume transferred during a transfer. There are two principal causes of transient errors. The first, and most likely cause, is that the system is fitting straight lines to transfers that 'taper' asymptotically. The second is that the spikes are another feature of the inverse modelling process.

The system can be considered to be macroscopic in nature because minute by minute accuracy is not required. Thus the existence of transient errors in short duration flow estimates can be tolerated. However, transient errors must be minimised in the estimation of continuous flow rates because any error in the continuous flow rate will result in an incorrect plutonium concentration estimate in the propagation simulation because of its affect on the process stage simulations.

3.2 Detection Strategy

The short-term diversion detection strategy also influences the design of the tools. Two possible signals may be used to detect disagreements, the first of which is the flow rate itself because this is usually maintained constant. The second is the short-term simulation error ($e = \hat{V} - \tilde{V}$). The production of relevant residual signals for detection is an important design factor for the volume analysis tools.

The importance of there being no transient errors in the continuous flow estimate has already been discussed. The average continuous volumetric flow rate can be estimated over the period of time when batches are not transferred. If it can be assumed that the plant is operating steadily then they can be assumed to be constant for the duration of the batch flow. This assumption is verified by simultaneously monitoring the non-active feeds into the process stage upstream of the receipt tank or monitoring the metering device out of a feed tank.

Using an observer, the corrections required to this average flow rate so that it 'matches' the volume profile are estimated. This correction signal can then be examined to see if there are any flows that may be indicative of an unauthorised movement of material.

Volumetric batch flow rate estimates are tolerant of transient errors and are easily obtained. In a three tank set comprising of receiving, buffer, and feeding tanks, there are two possible volumes that can be used to estimate the batch transfers: the export from one tank and the corresponding import into the next. It is preferable to use those estimates obtained by analysing the buffer tank volume as the analysis is simplified and it avoids compounding the errors in the short-term simulation.

In a buffer tank, since the flows used in the simulation will 'match' the volume profile, any simulation errors will be due to unauthorised transfers of material during the quiet periods. Thus diversions during batch transfers will result in errors in the feeding or receiving tank. It is anticipated that any diversion that overlaps a batch transfer will be visible in either or both the signals.

3.3 Flow Rate Estimation

In this section, the tools developed to estimate the volumetric import and export flows of both buffer and feeding/receiving tanks are described. These tools are constructed from the components outlined in Chapter 2. Note that since a feeding tank is essentially a reversed receiving tank, a single tool was developed for both.

Each tool is constructed from four components, the functions of which are:

1. Estimation of volume.
2. Detection of points of change.
3. Identification of points of change.
4. Calculation of flow rates.

Whenever a change is detected, the detector has to be suspended for a short period of time to allow the volume estimate to stabilise.

3.3.1 Buffer Tanks

This tool has been developed to estimate the volumetric import and export flows of buffer tanks. Estimation is based upon the changes in volume as material is transferred. This is a simple procedure if the transfer is to or from a buffer tank as opposed to a feeding/receiving tank. It remains a simple procedure if the duration of the transfer to/from a feeding/receiving tank is *always* relatively small. Note the emphasis is on the word 'always' since this is an operational decision. If this is not the case then a different approach is required; the development of which would require access to real plant data. Therefore this additional development work is outside the scope of this thesis.

The automatic estimation of a change in volume may not be that simple: noise might be present on the volume measurements and the transfer might 'taper' asymptotically (i.e. take a long time to finish). This would cause problems in ensuring that residual errors are minimised. Therefore it is unlikely that the most appropriate algorithm can be specified without reference to actual plant data. However, a data compression tool based upon these algorithms was shown to be adequate when applied to real measurement data from a product storage area (Appendix 1). The algorithm presented here was developed to analyse the simulated plant data and should be close to what is required.

3.3.1.1 Tool Design

The tool consists of four elements whose functions were outlined in the introduction to section 3.3. An RLS estimate of the volume is produced to filter the 'noise'. This estimate is delayed slightly to improve the detection of abrupt changes. Suitable change detectors are the V-mask or the Modified Shewhart Control Chart (MSCC). Both are equal in terms of performance although the MSCC is preferred on the basis of its simple design and tuning process.

Having detected the points (which, are necessarily after the change has occurred) it is necessary to identify start and stop volumes and the duration of the flow. The rectangle search (Section 2.5.3) is ideal as it can be tuned to ensure that a volume before and after the flow is selected. The accuracy of the time period is not so important as transient errors can be tolerated. The flow rate can then be found by averaging. If there is a desire to minimise transient errors, an observer could be designed to correct the estimated flow rates. The design of this observer would be similar to that discussed in Section 2.4. It is doubtful that the added complexity would bring any extra benefit.

Note that this tool is designed for operation in real-time, the calculation of the flow rates does not take place until a 'transfer pair' has been found.

3.3.1.2 Tool Performance

Figure 3.3.1.1 shows the start and end points of the volume transfers identified by the tool (denoted by a asterisk) superimposed on the volume measurement record of the buffer tank that was analysed. Figure 3.3.1.2 shows the simulation error of a normal plant for 2000 minutes. Note that transient errors exist, only the residual errors have been minimised. The performance of the tool meets the requirements outlined earlier.

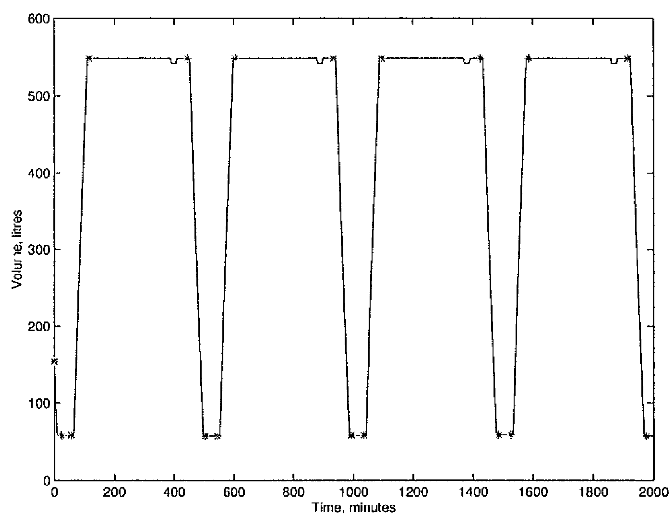


Figure 3.3.1.1 Start and end points of volume transfers identified by tool

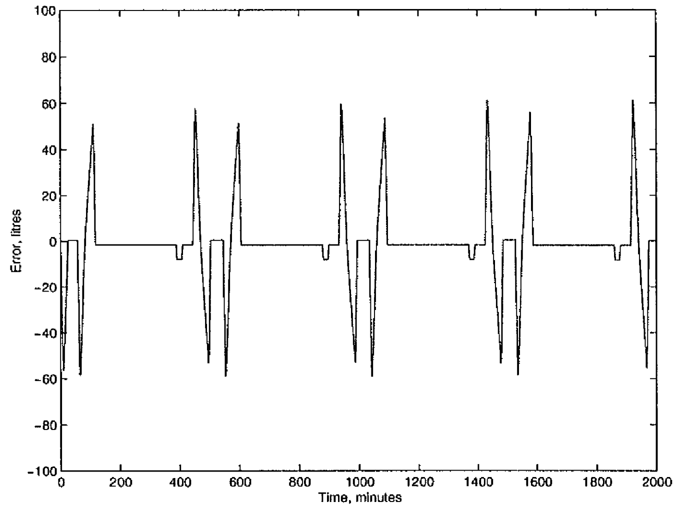


Figure 3.3.1.2 Simulation error, buffer tank

3.3.2 Estimation of Continuous flow rates.

This tool has been developed to estimate the continuous flow in feeding or receiving tanks. The estimate for the continuous flow rate into a tank (for a receiving tank) or out of a tank (for a feeding tank) is obtained by observing changes in its volume. This tool would operate in conjunction with the previous tool developed for buffer tanks, which would estimate the batch transfers, in the same tank. If metering device outputs were available on the continuous flow lines, this data could be used instead.

Again the tool consists of the four distinct parts described in the introduction to Section 3.3, which when combined estimate the flow rate. The first two parts are similar to those of the buffer tank tool: a RLS estimate to filter the 'noise' from the measurements and either a V-mask or MSCC to detect changes in the RLS estimate. The mode of operation of the detector differs slightly in this tool from that previously described. The detector is not only used to detect the points of change but also to identify the individual fill/empty cycles from the detected points. The first two parts operate continuously in real-time; the final two stages are invoked once a cycle for reasons that will be explained later.

The description here is for a receiving tank. The equations for a feeding tank would be very similar. Let the volume in the tank, at any time t, be V, the flow rate in be f_{in} and the flow rate out be f_{out}. Then:

$$dV/dt = f_{in} - f_{out} \qquad (3.1)$$

Over a single cycle it is normal for the flow rate into the tank to remain fairly constant so that, without loss of generality,

$$f_{in} = \alpha_i + \Delta f_{in}(t) \qquad (3.2)$$

where α_i is a constant over the cycle i. The output is a batch transfer:

$$f_{out} = \begin{cases} 0 & ; \ t \text{ otherwise} \\ g_{out}(t); & t : t_{s,i} < t < t_{f,i} \end{cases} \qquad (3.3)$$

where $t_{s,i}$ and $t_{f,i}$ are the times that the ith transfer commences and finishes. The purpose of the tool is to estimate f_{in}, the out flow rate g_{out} can also be obtained, but it is surplus to requirements because its estimation is less complicated from the adjoining buffer tank.

3.3.2.1 Estimation of α_i

The constant flow rate, α_i, is only estimated whilst f_{out} is assumed to be zero, it is assumed to remain constant when f_{out} is not equal to zero. Therefore, the key points of the cycle are $t_{f,i-1}$ (some time after the previous batch transfer has stopped), $t_{s,i}$ (the time at which the current batch transfer commences), and $t_{f,i}$ (the time at which the current batch transfer ends). In practice, a slightly later time $t_{s,i}^{+}$ is identified (Figure 3.3.2.1).

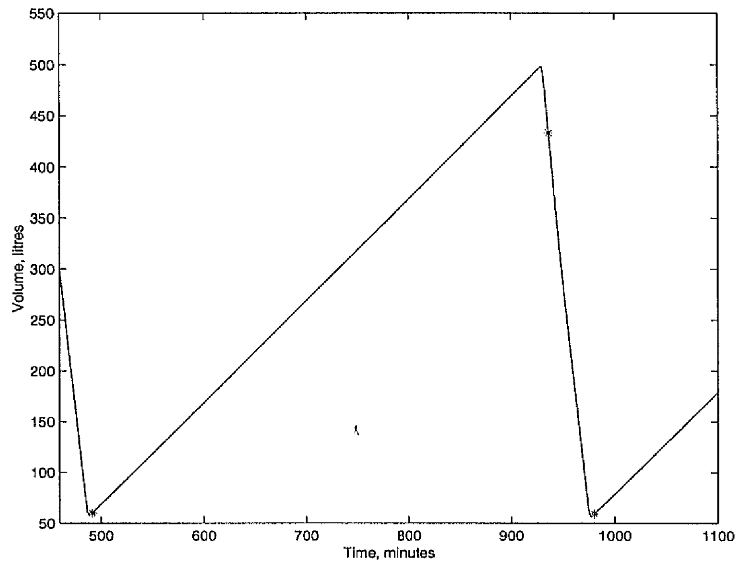


Figure 3.3.2.1 Fill/empty cycle

There are several methods to estimate α_i over the time period $t : t_{f,i-1} \le t < t^+_{s,i}$. The simplest method, averaging, is crude but effective. The flow rate is simply the gradient between some data point after $t_{f,i-1}$ and another data point some time before $t^+_{s,i}$ and $t_{s,i}$, when it is fairly certain that the batch transfer isn't taking place. This method produces satisfactory estimates.

A more sophisticated method avoids making this assumption by estimating parameters α_i, β and $t_{s,i}$ over the time period $t : t_{f,i-1} \le t < t^+_{s,i}$, where

$$\beta = \frac{\int_{t_{s,i}}^{t^+_{s,i}} g_{out}\, dt}{(t^+_{s,i} - t_{s,i})} \tag{3.4}$$

i.e. an average flow rate out, β, is assumed for the small time period $t : t_{s,i} \le t < t^+_{s,i}$. The volume tank model is then solved to provide volume measurements, $\widetilde{V}$, that can be compared with the measurements taken from the tank. Estimation is achieved by choosing those parameters (f_{in}, f_{out} and $t_{s,i}$) that minimise the square of the cumulative sum of the differences:

$$Errsum = \sum_{i=1}^{m} (\hat{V}_i - \tilde{V}_i) \qquad (3.5)$$

Simulated annealing is used to optimise these parameters. An estimate for g_{out} could then be obtained by numerical differentiation if so desired.

3.3.2.2 Estimation of $\Delta f_{in}(t)$

Having obtained estimates for α_i and f_{out} (from the buffer tank tool), $\Delta f_{in}(t)$, the correction term of the continuous flow in can be estimated. An observer is an ideal estimator in this instance. The observer equation is:

$$d\tilde{V}/dt = \alpha_i - f_{out} + K(\hat{V} - \tilde{V}) \qquad (3.6)$$

where: $K(\hat{V} - \tilde{V}) = \Delta f_{in}(t)$

α_i - constant term of continuous flow rate

f_{out} - batch flow out estimate

$\hat{V}$ - volume measurement

$\tilde{V}$ - volume estimate

The block diagram for this observer is shown in figure 3.3.2.1. Since α_i is the averaged constant flow rate over cycle i, $\Delta f_{in}(t)$ can be examined for short-term disagreements during the time period $t : t_{f,i-1} \le t < t_{s,i}$.

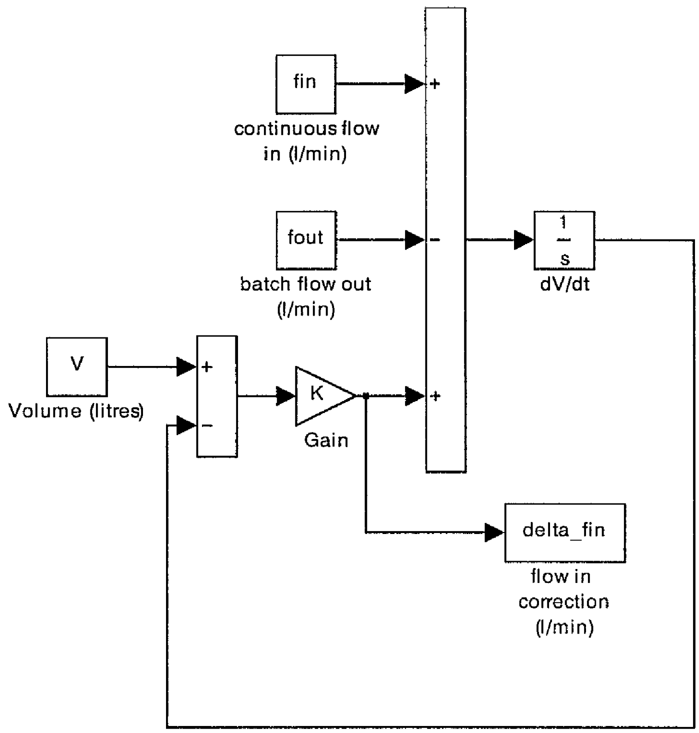


Figure 3.3.2.1 Observer for estimating $\Delta f_{in}(t)$

3.3.2.3 Tool Performance

Figures 3.3.2.2 to 3.3.2.5 show the output of the tool when a normally operated receiving tank is analysed.

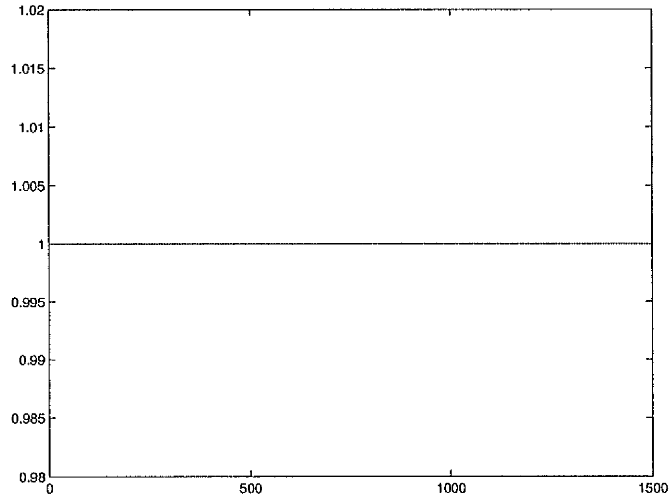


Figure 3.3.2.2 α_i estimate (l/min)

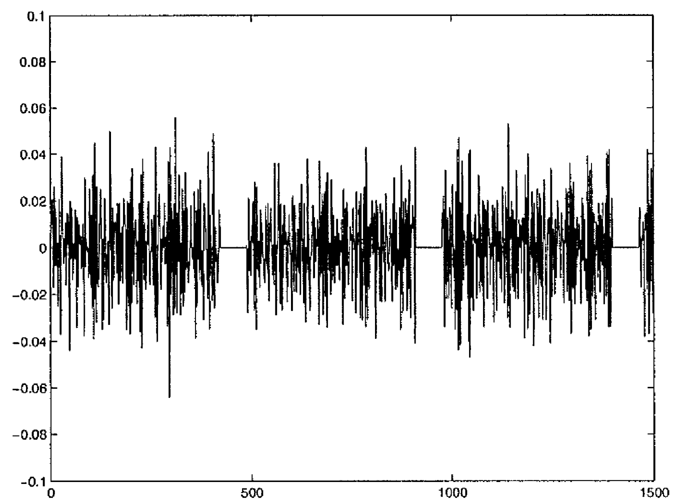


Figure 3.3.2.3 $\Delta f_{in}(t)$ estimate (l/min)

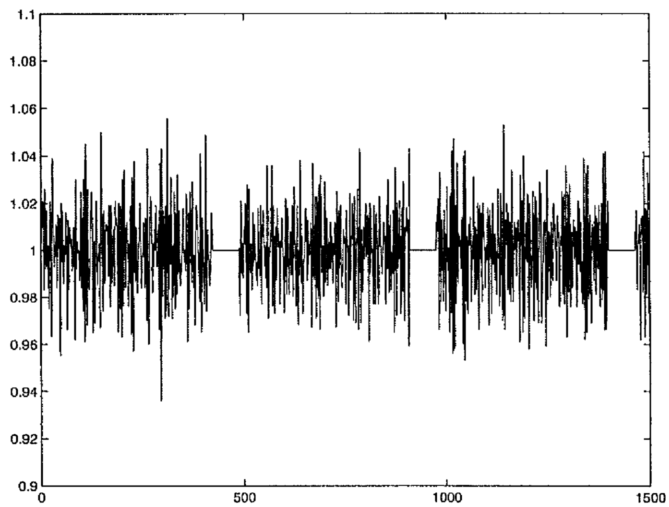


Figure 3.3.2.4 f_{in} estimate (l/min)

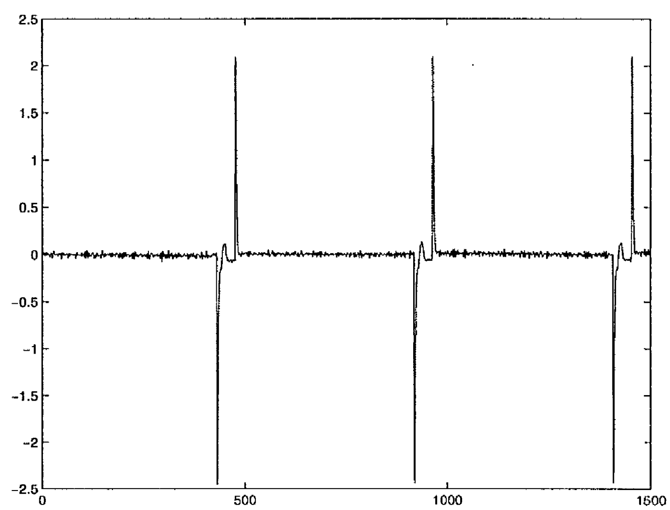


Figure 3.3.2.5 Simulation error signal (litres)

The transient spikes in Figure 3.3.2.5 are due to the mismatch between the batch transfer estimated by the buffer tank tool and the actual flow. As these errors are transient (the net effect is zero), they can be ignored.

3.4 Short-term Disagreement Detection

As explained in section 3.2, two signals are monitored for possible disagreements:

1. The estimate of the correction to the continuous flow rate.
2. The short-term simulation error signal for feeding/receiving tanks for disagreements in batch transfer estimates.
3. The short-term simulation error signal for disagreements that occur when there are no normal movements on a buffer tank.

As all of these signals are time-series of mean zero, a suitably tuned standardised cumulative sum is an appropriate detector.

The tolerances of the cusum were set sufficiently large so that the desired level of sensitivity was obtained. The detectors extract the 'start' and 'stop' times and the time history pertaining to the incident. Once a disagreement has been detected, a sub-event is created to represent the disagreement.

3.5 Short-term Disagreement Diagnosis

There are only three possible explanations for negative changes in the continuous flow rate: Three possible explanations for changes in the continuous flow rate are:

1. liquor has been transferred to hidden inventory (i.e. removed from the tank);
2. the acid molarity and the Pu concentration have changed simultaneously;
3. the unspecified component and Pu concentration have changed simultaneously.

Simultaneous actions might be required to satisfy both volume and density measurements.

To diagnose short-term disagreements, a set of observers is invoked concurrently to quantify the three possible solutions by analysing the time history extracted by the detector. Further observers can be added to expand the possible explanations for the disagreement. Descriptions and scores can then be attached to each of the three possibilities, which are then attached to the sub-event as diagnoses. The transfer to hidden inventory is chosen preferentially as it is a single parameter solution.

Similarly there are four possible explanations for disagreements in the short-term simulation error:

1. liquor has been transferred to hidden inventory (i.e. removed from the tank);
2. plutonium nitrate solution of similar concentration has been added to the tank;
3. plutonium nitrate solution of dissimilar concentration has been added to the tank;
4. acid has been added to the tank.

Again, descriptions and scores can then be attached to each of these possibilities, which are then attached to the sub-event as diagnoses. The transfer to hidden inventory is chosen preferentially as it is a single parameter solution, especially as the others are not physically meaningful if material is leaving the tank.

CHAPTER 4

DENSITY MEASUREMENT ANALYSIS

4.0 Introduction

In this chapter the tools developed to analyse the density measurements are described. The system is designed to propagate the plutonium through the plant using a simulation. The start point of the simulation is the input accountancy tank where the composition of the liquor is known, the individual components can be traced through the plant to the first process stage (cycle 1). This can be achieved by adding mass balances pertaining to acid, uranium and unspecified (for all other components) to the simulation. In theory, plutonium concentration can now be confirmed by comparing the density calculated on the basis of Equation 1.4 with the density measurements. A similar process can be applied after the first cycle, and so on, provided that the composition of the liquor can be assumed at the start of each set of tanks.

Unfortunately the composition of liquor exported by a cycle (i.e. imported into the receiving tank) is unlikely to be known accurately for various operational reasons. It is therefore difficult to attribute fluctuations in density to fluctuations in plutonium concentration because density might fluctuate as a result of changes in the concentrations of any of the other components of the liquor. As a consequence of this detection must be relatively insensitive and diagnosis must accommodate the various possibilities.

With reference to section 1.6, this chapter contributes activity 2 in its entirety, the analysis of density measurements using inverse modelling. It also contributes towards activities 3 and 4, the short-term disagreement detection and diagnosis activities. Of these contributions, all are original.

4.1 Density Measurement

Before describing the tools, it is important to appreciate that density measurements are just that. At a given temperature (T), the density ρ_s, of any solution, which is likely to be present, can be calculated on the basis of a correlation like Equation 1.4, which is repeated here:

$$\rho_s(T) = \rho_w(T) + \alpha_{Pu}(T)[Pu] + \alpha_{H+}(T)[H+] + \text{Equivalent terms associated with other elements} \tag{4.1}$$

Where: $\rho_w(T)$ - density of water (g/l)

$[Pu]$ - Pu concentration (g/l)

$[H+]$ - Acid molarity (mol/l)

$\alpha_{Pu}(T)$ & $\alpha_{H+}(T)$ - coefficients

For the purpose of the analysis, a shortened version of this equation is used to accommodate the fact that many of the individual components are not known. An additional term, $\alpha_{unsp}(T)[Unsp]$, is introduced to represent the contribution to the density of the liquor of these unspecified components. The density correlation is now:

$$\rho_s(T) = \rho_w(T) + \alpha_{Pu}(T)[Pu] + \alpha_{H+}(T)[H+] + \alpha_{unsp}(T)[Unsp] \tag{4.2}$$

At a temperature of 25 degrees Celsius, and with no unspecified components in the liquor, the density correlation is evaluated to be (Howell and Scothern, 1995):

$$\rho_s = 997.02 + 1.47[Pu] + 34[H+] \tag{4.3}$$

Thus if the Pu concentration is known from the propagation simulation, the acid molarity can be estimated.

If the liquor density, acid molarity, unspecified concentration, and temperature of the liquor are known then the concentration of plutonium can be obtained. However if only a density measurement is available, and without making any further assumptions, the individual components cannot be estimated separately.

The inseparable nature of the liquor constituents means that the term X is introduced to denote this fact:

$$\rho_s(T) = \rho_w(T) + X \tag{4.4}$$

Changes in X are indicative that one or more of the components of the liquor are changing.

The tool consists of three components:

1. Estimation of process stage output X concentration
2. Detection of changes in X concentration
3. Diagnosis of changes in X concentration.

4.2 Analysis Tools

4.2.1 Estimation of Process Stage output X concentration

The function of this sub-tool is to estimate the X concentration of the import flow into the receiving tank. The mass of X in the receiving tank is given by:

$$dMx/_{dt} = fx_{in} - fx_{out} \tag{4.5}$$

where: Mx - the mass of X in the tank (g)

fx_{in} - the flow of X into the tank (g/min)

fx_{out} - the flow of X out of the tank (g/min)

As the liquor within the tank can be assumed to homogeneous, the X concentration of the export flow of the tank is equal to the X concentration of the tank itself and so the above equation can be rearranged as:

$$dMx/_{dt} = f_{in}[X_{in}] - f_{out}[X_{tk}] \tag{4.6}$$

where: Mx - the mass of X in the tank (g)

f_{in} - volumetric flow into tank (l/min)

f_{out} - volumetric flow out of tank (l/min)

X_{in} - X concentration into tank (g/l)

X_{tk} - X concentration in tank (g/l)

The density of X in the tank can be obtained from the density and temperature measurements, and so it follows that the mass of X in the tank is relatively easy to obtain. Both the volumetric import and export flows are available from the tools described in Chapter 3. Using the above equations, the X concentration can be estimated using an observer:

$$d\tilde{M}x/_{dt} = K(\hat{M}x - \tilde{M}x) - f_{out}[X_{tk}] \tag{4.7}$$

where: $$K(\hat{M}x - \tilde{M}x)/_{f_{in}} = X_{in}$$

$\hat{M}x$ -measured mass of X in the tank

$\tilde{M}x$ -estimated mass of X in the tank

The block diagram for this observer is shown in figure 4.2.1.

4.2.2 X Concentration Disagreement Detector

In normal operation, the X density exported by the process stage remains close to a constant value with minor fluctuations e.g. slop on the outflow from the cycle or minor variations in the solvent feed concentration. The detector has to ignore these minor deviations and alarm when a major incident occurs. An appropriate detector is the standardised cumulative sum.

The tolerances of the cusum were set sufficiently large so that the desired level of insensitivity was obtained. The detector extracts the 'start' and 'stop' times and the time history pertaining to the incident. It then creates a sub-event to represent the disagreement.

4.2.3 Diagnosis

As illustrated in section 4.1, fluctuations in density may be caused by fluctuations in the concentration of any of the components of the liquor. The concentrations of all three components (plutonium, acid, and unspecified) are known before the incident occurs. Therefore, by 'freezing' two of the components, the change required in the third component to produce the fluctuation in density could be estimated.

The three equations required, obtained by rearranging the standard density correlation, are:

$$[Pu] = \frac{X - \alpha_{H+}(T)[H+] - \alpha_{unsp}(T)[unsp]}{\alpha_{Pu}(T)} \tag{4.8}$$

$$[H+] = \frac{X - \alpha_{Pu}(T)[Pu] - \alpha_{unsp}(T)[unsp]}{\alpha_{H+}(T)} \tag{4.9}$$

$$[unsp] = \frac{X - \alpha_{Pu}(T)[Pu] - \alpha_{H+}(T)[H+]}{\alpha_{unsp}(T)} \tag{4.10}$$

These three diagnostic tools can be invoked concurrently. Descriptions and scores can then be attached to each of the three possibilities. These possibilities are then attached to the sub-event as diagnoses.

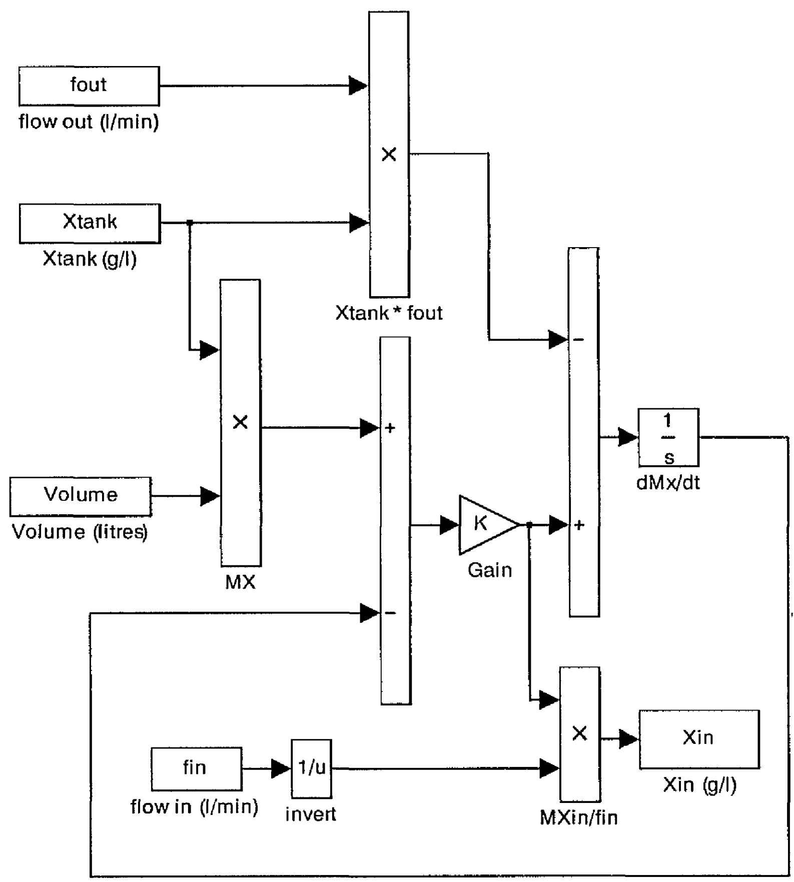


Figure 4.2.1 X in concentration observer

CHAPTER 5

MODEL BASED REASONING

5.0 Introduction

Having completed the analysis of the volume and density measurements, all the information required by the system for it to provide medium-term assurances is now available. The detection and evaluation of disagreements over the medium-term is different because the emphasis is on the development of a gradual mismatch between the predicted and measured material distribution.

An initial estimate of the plutonium distribution within the plant can be obtained by propagating material through the plant from the inlet accountancy tank using a simulation. There will always be a gradual mismatch between the predicted distribution and the measured inventory, even when the plant is operating normally. A gradual mismatch can derive from:

1. flow rate estimation errors;
2. measurement biases;
3. a plant incident.

The term redistribution is then used to describe the process of moving plutonium around the plant simulation so that it predicts that measured. Other plant information is required to choose which of the above has caused this redistribution.

To compensate for this gradual divergence, and at the same time, to be in a position to detect and diagnose an event that occurs gradually, the entire plant is analysed as one. That is the focus is macroscopic rather than microscopic. The reason for this is that the amount of material involved during a gradual event is negligible when analysed on a per cycle basis. Only when the plant is examined as a whole over a long period of time does the gradual divergence become apparent. To achieve an agreement, a tool is required to redistribute plutonium over time.

Measurement biases can affect both the flow rate estimates and the comparison directly. Having obtained estimates for the biases the difficulty then is to decide whether a new 'mismatch' is due to a change in biases or either of the other two options. If the inspectors have confidence in the accuracy of the bias estimates, i.e. the system has established a record of good performance and if the redistribution is relatively large, then the redistribution would be attributed to real movements. However, for the plant operators, the possibility of the existence of a bias is the preferred explanation for a medium-term disagreement. Supporting evidence should also be taken into consideration, such as the recent replacement or re-calibration of instrumentation.

With reference to section 1.6, this chapter contributes activities 5,6, and 7, of which all are original. This chapter is split into three sections: the first section describes the models of the plant used within the system; the second describes the method developed for the estimation of gross systematic multiplicative biases; the redistribution and medium-term detection and diagnosis tools are described in the final section.

5.1 Plant Models

The chemical separation area is not modelled as a whole for reasons of computational simplicity. The process stages provide natural boundaries at which to separate the plant into segments comprising of tank-sets or process stages. Each segment is simulated separately and comprises of one or more discrete units with zero-inventory connections. If, on the real plant, the assumption of zero-inventory pipe models is felt to be inadequate, then simple pipe inventory models could be included. The outputs of the individual simulations are combined before being analysed as a whole.

Two types of models exist, one for tanks and one for process stages. Process stages are also known as 'hidden inventories' as their plutonium inventories are unmeasured.

5.1.1 Tank Models

The tank model is composed of a number of material balances like:

$$dM_{Pu}/dt = f_{in}[Pu]_{in} - f_{out}[Pu] \qquad (5.1)$$

where: M_{Pu} - mass of Pu in the tank (grams)

f_{in} - volumetric flow into the tank (l/min)

f_{out} - volumetric flow out of the tank (l/min)

$[Pu]_{in}$ - Pu concentration of inlet (g/l)

$[Pu]$ - Pu concentration of tank (g/l)

All of the tank models are based upon similar material balances. More complicated models are available, Howell & Scothern (1995), but again this is an implementation issue.

5.1.2 Process Stage Models

The choice of process stage model is again an implementation issue. From a safeguard's point of view, it is important to appreciate that the models are only needed to predict the Pu inventory within the individual stages and to predict the concentrations of Pu leaving the process stage. The units are fed from, and output to, tanks which 'smooth' out short-term fluctuations. Therefore, the system does not require accurate dynamic estimates of the operational status of individual locations within the process stage. However, more accurate models may be used, but the extra benefit the added complexity will bring is difficult to determine.

The solvent extraction cycle is based on the steady state follower approach. This model involves the formation of pairs of equations, the first to accommodate changes in the process flow sheet and the second to accommodate deviations from the nominal inventory value. The model of the plutonium inventory is thus:

$$\tau \, dI/dt + I = I_{nom} \qquad (5.2)$$

$$d(I + I_{hinv})/dt = f_{in}[Pu]_{in} - f_{out}[Pu]_{out} \qquad (5.3)$$

where: I_{nom} - nominal Pu inventory of cycle (grams)

I - current Pu inventory of cycle (grams)

τ - time constant of cycle

I_{hinv} - Pu inventory of hidden inventory (grams)

f_{in} - volumetric flow into the cycle (l/min)

f_{out} - volumetric flow out of the cycle (l/min)

$[Pu]_{in}$ - Pu concentration of inlet (g/l)

$[Pu]_{out}$ - Pu concentration of outlet (g/l)

The time constant of the process stage, τ, is determined during commissioning, and need only be known approximately. A similar pair of equations is used to model the uranium inventory in the first solvent extraction cycle. If the plant is assumed to be operating in accordance with the flow sheet, then the rate of change of the deviation inventory is zero. Thus the model now becomes:

$$[Pu]_{out} = \frac{f_{in}[Pu]_{in} - {}^{dI}\!/_{dt}}{f_{out}} \tag{5.4}$$

Therefore, the flow rate of plutonium leaving the cycle can be calculated provided that the volumetric flow rate out is in accordance with the flow sheet. Otherwise, the first pair of equations needs to be solved. A similar approach holds for uranium. In the absence of a more sophisticated estimator, Candy & Rozsa (1980), a similar model can be adopted for the concentrator.

5.1.3 Distribution Model

Having obtained the estimates for the inter-tank volume transfers, it is a straightforward task to distribute the plutonium throughout the plant using a suitable simulation. The preferred simulation has both volume and Pu balances, but the implementers may wish to include other balances, e.g. mass of acid.

The volumetric flow rate and plutonium concentration of the flow into the first tank of the plant is known from the accountancy vessel upstream. Thus it is a straightforward task to simulate the flow of material through the first tank set to the first process stage. It is assumed that the inventory of the all the process stages is made available by the operators, so the plutonium concentration output by the process stage can be calculated.

The simulation continues on through the next set of tanks and so on until the final tank has been reached. The export flow and plutonium concentration of the final tank is known from the product accountancy tank.

5.2 Estimation of Gross Systematic Multiplicative Biases

Modelling and identifying gross errors in process data is established in both steady-state and dynamic systems. For steady-state systems, several methods based upon linear data-reconciliation have been proposed. These methods are known as the 'global test' (Reilly and Carpani, 1963), the 'nodal test' (Reilly and Carpani, 1963; Mah et al., 1976), the 'measurement test' (Mah and Tamhane, 1982; Crowe et al., 1983), 'generalised likelihood ratios' (Narasimhan and Mah, 1987), etc. The situation is somewhat different for dynamic systems, the application of data-reconciliation is still in its infancy (Liebman et al., 1992; Sistu et al., 1993).

Those schemes that have been developed for dynamic systems (Bagajewicz and Jiang, 1998; Sanchez et al., 1999; Albuquerque and Biegler; 1996) are reliant upon the availability of measured flow rate variables or flow sheets for the plant. Another drawback is that all the measurements required for the application to be successful must be available. This is not the case with reprocessing plants.

A general method is proposed for isolating and estimating gross biases by analysing the predicted movements of both bulk and nuclear material over time. The foundation for the method is based on an examination of how a bias would affect the flow rate calculations for either a buffer or feed/receipt tank, and how these biases therefore affect tank volume predictions over time. Equations can be derived to show how these predictions would diverge from the volume measurements and how biases would cause the predicted plutonium inventory in the product accountancy tank to diverge from that measured. Given these understandings, a method is then proposed to solve the inverse: 'if disagreements are observed between measurements and predictions, what are the biases?'

For simplicity the description is based on a three tank-set comprising of a receipt tank, a buffer tank, and a feed tank. In the plant model, the estimates for the short duration transfers from the buffer tank analysis are utilised. Reference is made to other types of tank-sets where appropriate.

The following subscript convention is used throughout the analysis. An identifier, 'b', 'f', or 'r' denotes the type of tank (buffer, feed, or receipt). This identifier maybe followed by a number if more than one of that type of tank exists in the tank set. For example, V_{b2} refers to the volume of buffer tank number two.

5.2.1 Affects of Gross Multiplicative Biases on the Plant

If the tank measurements are biased, then this will bias the transfer estimates and hence the model of how nuclear material is progressing through the plant. The estimation of biases that affect the density measurements is not as important to the overall performance of the system. The effect of a biased density measurement will be to over or underestimate the acid concentration input to a receipt tank. If only one tank is estimated, its effect will be fairly transparent so that the bias can be isolated and estimated in a straightforward manner; if not, estimation is more complicated.

5.2.1.1 Buffer Tanks

Consider some arbitrary operational cycle, k, of a buffer tank: in the beginning the tank has initial volume V_L^1, is then filled to volume V_H^1. Sometime later, the tank is emptied from volume V_H^2 to volume V_L^2. If these volumes are measured, and these measurements are subject to a multiplicative bias, ε_b, which is assumed to be constant over the time period of the analysis, then the total volume input and output during cycle k ($\widetilde{I}_k$ & $\widetilde{E}_k$) can be estimated as:

$$\widetilde{I}_k = V_H^1(1+\varepsilon_b) - V_L^1(1+\varepsilon_b) = I_k + \varepsilon_b I_k \qquad (5.5)$$

$$\tilde{E}_k = V_H^2(1+\varepsilon_b) - V_L^2(1+\varepsilon_b) = E_k + \varepsilon_b E_k \qquad (5.6)$$

These equations do not take into consideration any other direct additions/withdrawals of liquor. It is assumed that the system would account for these activities. If evaporation is assumed to insignificant and if any sampling results in a relatively small change in volume then, assuming the estimates for $\tilde{I}_k$ & $\tilde{E}_k$ are accurate:

$$\tilde{V}_L^2 = \hat{V}_L^1 + \tilde{I}_k - \tilde{E}_k = \hat{V}_L^2 \qquad (5.7)$$

If, over a period of time, the buffer tank performs N fill/empty cycles then:

$$Total\ input = \sum_{k=1}^{N} \tilde{I}_k = \sum_{k=1}^{N} I_k + \varepsilon_b \sum_{k=1}^{N} I_k \qquad (5.8)$$

$$Total\ Output = \sum_{k=1}^{N} \tilde{E}_k = \sum_{k=1}^{N} E_k + \varepsilon_b \sum_{k=1}^{N} E_k \qquad (5.9)$$

5.2.1.2 Receipt/feed Tanks

A receipt/feed tank either import continuously and export in batches or vice versa. The analysis presented here is for a receipt tank, the analysis for a feed tank is similar. For a single cycle, k, over the period of time ($t_{k,s}$ to $t_{k,e}$) when the output batch is not transferred, the average continuous volumetric flow rate, $\bar{f}_k$, can be estimated on the basis of the volume measurements recorded (subjected to multiplicative bias ε_r):

$$\tilde{\bar{f}}_k(t_{k,e} - t_{k,s}) = \int_{t_{k,s}}^{t_{k,e}} \frac{d\hat{V}_r}{dt} dt = \int_{t_{k,s}}^{t_{k,e}} \frac{dV_r}{dt} dt + \varepsilon_r \int_{t_{k,s}}^{t_{k,e}} \frac{dV_r}{dt} dt \qquad (5.10)$$

As the batch volume input/output is estimated on the observed changes in the buffer tank, the continuous flow rate can now be estimated over the entire fill/empty cycle ($t_{k,s}$ to $t_{k,f}$). The 'true' volume at the end of a cycle is given by:

$$V_{r_f} = V_{r_s} + \int_{t_{k,s}}^{t_{k,f}} \frac{dV_r}{dt} dt - E_k \tag{5.11}$$

And the estimated volume at the end of the cycle is given by:

$$\widetilde{V}_{r_f} = \hat{V}_{r_s} + \widetilde{\bar{f}}_k (t_{k,f} - t_{k,s}) - \widetilde{E}_k \tag{5.12}$$

Which can be re-arranged to:

$$\widetilde{V}_{r_f} = \hat{V}_{r_s} + \int_{t_{k,s}}^{t_{k,f}} \frac{dV_r}{dt} dt + \varepsilon_r \int_{t_{k,s}}^{t_{k,f}} \frac{dV_r}{dt} dt - E_k - \varepsilon_b E_k \tag{5.13}$$

Note the re-arranged equation distinguishes between the two errors because the biases pertain to different tanks. The equation can be extended to multiple fill/empty cycles (over the time period t_s to t_f, the duration of N cycles) using the equations for total input and output from the buffer tank analysis:

$$\widetilde{V}_{r_f} = \hat{V}_{r_s} + \int_{t_s}^{t_f} \frac{dV_r}{dt} dt + \varepsilon_r \int_{t_s}^{t_f} \frac{dV_r}{dt} dt - \sum_{k=1}^{N} E_k - \varepsilon_b \sum_{k=1}^{N} E_k \tag{5.14}$$

For a feed tank, this equation is:

$$\widetilde{V}_{f_f} = \hat{V}_{f_s} + \sum_{k=1}^{N} I_k + \varepsilon_b \sum_{k=1}^{N} I_k - \int_{t_s}^{t_f} \frac{dV_f}{dt} dt - \varepsilon_f \int_{t_s}^{t_f} \frac{dV_f}{dt} dt \tag{5.15}$$

5.2.1.3 Tank-set bias equations

Volume estimates derived on the basis of the above are now compared with the measured volumes over a number of cycles. For a tank-set composed of three tanks, the disagreements would be as follows:

Receipt tank: $\delta_r = \hat{V}_{r_f} - \tilde{V}_{r_f} = \varepsilon_r (V_{r_f} - V_{r_s}) - \varepsilon_r \int_{t_s}^{t_f} \frac{dV_r}{dt} dt + \varepsilon_b \sum_{k=1}^{N} E_k$ (5.16)

Buffer tank: $\delta_b = \hat{V}_{b_f} - \tilde{V}_{b_f} = 0$ (5.17)

Feed tank: $\delta_f = \hat{V}_{f_f} - \tilde{V}_{f_f} = \varepsilon_f (V_{f_f} - V_{f_s}) + \varepsilon_f \int_{t_s}^{t_f} \frac{dV_f}{dt} dt - \varepsilon_b \sum_{k=1}^{N} I_k$ (5.18)

Consider the receipt tank, if either $\varepsilon_r >> \varepsilon_b$ or $\varepsilon_b >> \varepsilon_r$ then the measured and estimated final volumes will diverge in time. A similar situation arises for feed tanks. Note also that a bias in the buffer tank will affect both the feed and receipt tank equations, but not the equation for the buffer tank. If all the biases are similar, there will be no increase. An additional procedure has to be included to accommodate this eventuality.

It is important to appreciate that there are only certain instances when $\tilde{V}$ can be estimated sensibly because of the assumptions about the continuous flow rates, evaporation etc. This limits the number of 'points of disagreement' that are available for the analysis.

If only a single set of measurements were available for a three tank-set then:

$$\begin{bmatrix} \delta_r \\ \delta_f \end{bmatrix} \equiv \begin{bmatrix} \hat{V}_{r_f} - \tilde{V}_{r_f} \\ \hat{V}_{f_f} - \tilde{V}_{f_f} \end{bmatrix} = AD_3 \begin{bmatrix} \varepsilon_r \\ \varepsilon_b \\ \varepsilon_f \end{bmatrix} \tag{5.19}$$

where:

$$A = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}$$

$$D_3 = \begin{bmatrix} g_r & 0 & 0 \\ 0 & g_b & 0 \\ 0 & 0 & g_f \end{bmatrix}$$

$$g_r = \int_{t_s}^{t_f} \frac{dV_r}{dt} dt - (\hat{V}_{r_f} - \tilde{V}_{r_s})$$

$$g_b = \sum_{k=1}^{N} I_k$$

$$g_f = \int_{t_s}^{t_f} \frac{dV_f}{dt} dt - (\hat{V}_{f_f} - \tilde{V}_{f_s})$$

Biases in other tank sets can be modelled in the same way. For instance for a four tank-set composed of three buffer tanks $(b1, b2, b3)$ and a feed tank (f), where the first tank receives the input accountancy flow, the model becomes:

$$\begin{bmatrix} \delta_{b1} \\ \delta_{b3} \\ \delta_f \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} D_4 \begin{bmatrix} \varepsilon_{b2} \\ \varepsilon_{b3} \\ \varepsilon_f \end{bmatrix} \tag{5.20}$$

If transfers from the input accountancy tank are accurately known and the flow rate estimate out of tank b1 and into tank b2 is based on $\hat{V}_{b2}$ and so on. The equivalent matrix A is square because the equation pertaining to the first buffer tank only refers to one bias because the input accountancy tank multiplicative bias is assumed to be negligible. It is then a straightforward calculation to estimate ε_{b1}.

5.2.1.4 The Entire Plant

For the purposes of this analysis, the concentrator and solvent extraction cycles can be viewed as transformers with physical inventories:

$$\frac{dM_{Pu}}{dt} = f_f[Pu]_f - f_r[Pu]_r \tag{5.21}$$

where [] denotes a concentration and f a flow. The outlet conditions are estimated on the basis of knowledge of the instantaneous nominal inventory, $M_N(t)$:

$$\frac{dM_N}{dt} = \tilde{f}_f[\tilde{P}u]_f - \tilde{f}_r[\tilde{P}u]_r \tag{5.22}$$

If either or both of the flow rate estimates are derived from biased volume measurements, then:

$$[\tilde{P}u]_r = \frac{(1+\varepsilon_f)\frac{dV_f}{dt}[\tilde{P}u]_f - \frac{dM_N}{dt}}{(1+\varepsilon_r)\frac{dV_r}{dt}} \tag{5.23}$$

Therefore the plutonium concentration out will be over or underestimated depending on the relative values of the feed and receipt tank biases. This over/underestimate will be propagated downstream of the cycle, possibly being compounded because of errors in other feed/receipt tanks.

The error will be detected when the material reaches the product accountancy tank because material transfers are measured at this KMP. That is the physical inventory estimate for this tank will continue to rise or fall because of the biased estimate in plutonium concentration:

$$\frac{d\tilde{M}_{Pu}}{dt} = \sum_{k=1}^{N_i}[\tilde{P}u]_k \tilde{I}_k - \sum_{k=1}^{N_o}[\hat{P}u]_k \hat{E}_k \tag{5.24}$$

Thus the residual ($\hat{M}_{Pu} - \tilde{M}_{Pu}$) for the product accountancy tank is an important indicator of one or more biased tanks upstream.

5.2.2 Identifying the gross biases

To identify the gross biases, a systematic search procedure is used. The biases are estimated on the bias of equations like those in section 5.2.1.3.

Starting at the product accountancy tank, the gross biases for each tank-set are estimated one tank-set at a time. Any biases that are estimated are used immediately to revise the plutonium concentration downstream of the tank-set. This results in a new value of the residual ($\tilde{M}_{pu} - \hat{M}_{pu}$) for the product accountancy tank.

The residual is a key indicator as to the status of the analysis. If, at any stage, the residual deviates significantly from zero, then either the biases in the current tank-set have been estimated incorrectly or the problem lies in a tank-set downstream of that being analysed.

A disadvantage of this method is that the residual is only one variable and it can be affected by biases in a number of different tanks. Additional information can be obtained by repeating the analysis on plant data collected subsequently, and by taking samples in 'suspect' tank-sets. This allows the comparison of the models with the 'true' throughput of the tank-set.

The method used to estimate the biases is dependent upon the status of matrix A. The matrix A may be defined, over-defined or under-defined. Solving for anything other than an under-defined A matrix is simple. If matrix A is square and invertible (i.e. $A \in R^m * R^m : rank(A) = m$), then the equation can simply be solved if $D = \tilde{D}$.

If A is over-defined (i.e. $A \in R^m * R^n, m > n : rank(A) = n$), then linear regression is an appropriate method for solving for the biases. Linear regression for w sets of measurements, is based on the following equation:

$$e = [X^T C^{-1} X]^{-1} X^T C^{-1} \begin{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_3 \end{bmatrix}_1 \\ \vdots \\ \begin{bmatrix} \delta_1 \\ \delta_3 \end{bmatrix}_w \end{bmatrix} \tag{5.25}$$

where $X = \begin{bmatrix} A\tilde{D}_1 \\ \vdots \\ A\tilde{D}_w \end{bmatrix}$ and C is the measurement covariance matrix, which is diagonal

and contains elements $\varepsilon^2_{random} \hat{V}^2$ where ε_{random} is the random error.

The assumption that $D = \tilde{D}$ can then be ameliorated by incorporating the bias estimates into the simulation to obtained a revised $\tilde{D}$, which can then be used to produce revised bias estimates and so on until convergence.

Otherwise A is under-defined and any attempt to perform linear regression would result in a singularity because the solutions are not unique. Instead only those biases that are likely to be significantly in error are identified and incorporated into the equations, on which linear regression may take place.

Consider a three tank-set, larger tank sets can be treated in the same way, but the analysis presented here is for a three tank-set. There exist seven possible combinations of gross biases ($\{\varepsilon_r\},\{\varepsilon_b\},\{\varepsilon_f\}$ $\{\varepsilon_r,\varepsilon_b\},\{\varepsilon_r,\varepsilon_f\},\{\varepsilon_b,\varepsilon_f\}$and $\{\varepsilon_r,\varepsilon_b,\varepsilon_f\}$), the last of which leads to singularity. Thus the combination of $\{\varepsilon_r,\varepsilon_b,\varepsilon_f\}$ results in non-unique solutions which require a different approach from those combinations that result in unique solutions. The aim is to identify those sub-sets that are likely to contain gross biases.

5.2.2.2 Under-defined A matrix – unique solutions

The algorithm is based on qualitative reasoning (Forbus (1984) and De Kleer & Brown (1984)), which is applied to matrix A and hence to only one set of measurements i.e. one disagreement per tank.

A gross bias is defined as a bias that results in a significant disagreement. A tolerance is set for each tank, and a gross bias is deemed to exist if the modulus of the disagreement exceeds this tolerance. The disagreement can then be classified as either +ve (i.e. the disagreement > tolerance), -ve (i.e. the disagreement < tolerance), or null (i.e. no disagreement).

If each bias in turn is taken to be non-gross, matrix A becomes square and thus invertible. Each of these three inverses can now be applied to the qualitative vector of disagreements on the basis of the quantitative/qualitative product operator defined in table 5.2.2.1. This results to the three sets of results shown in table 5.2.2.2. Note that 'u' means undefined whilst 'u0' means that although it is undefined, it is most likely to be zero. Similarly, a 'u+' means that although it is undefined, it is most likely to be positive.

Adopting the format $\{\varepsilon_r,\varepsilon_b,\varepsilon_f\}$, the sub-sets are then $\{\varepsilon_r,0,\varepsilon_f\}$, $\{\varepsilon_r,\varepsilon_b,0\}$, $\{0,\varepsilon_b,\varepsilon_f\}$ together with $\{\varepsilon_r,0,\varepsilon_f\}\,^\wedge\{\varepsilon_r,\varepsilon_b,0\}\,^\wedge\{0,\varepsilon_b,\varepsilon_f\}$, where '^' is applied along the three elements pertaining to ε_r and so on. For example, if both disagreements (δ_r and δ_f) are positive, then {-,0,+}, {-,-,0}, and {0,+,+} are obtained from the table 5.2.2.2, which when combined gives the fourth sub-set {u-,u,u+}.

		Qualitative values		
	*	-	0	+
Quantitative multiplier	-1	+	u0	-
	0	0	0	0
	1	-	u0	+

Table 5.2.2.1 Quantitative/qualitative product operator

		δ_3				δ_3		
	ε_r	-	0	+	ε_f	-	0	+
δ_1	-	+	+	u	-	-	u0	+
	0	u0	u0	u0	0	-	u0	+
	+	-	-	-	+	-	u0	+

		δ_3				δ_3		
	ε_r	-	0	+	ε_b	-	0	+
δ_1	-	+	+	u	-	+	u0	-
	0	+	u0	-	0	+	u0	-
	+	u	-	-	+	+	u0	-

		δ_3				δ_3		
	ε_b	-	0	+	ε_f	-	0	+
δ_1	-	-	-	-	-	-	-	u
	0	u0	u0	u0	0	-	u0	+
	+	+	+	+	+	u	+	+

Table 5.2.2.2 Disagreements to biases for 2-element sub-sets

Note that 'no disagreements' is not an indicator that no biases exist on the tank-set. For instance, the case where all three biases are positive and of similar magnitude might generate no disagreements. However, such cases would lead to a disagreement in the product accountancy tank and this feature is exploited in the method for non-unique solutions.

A parsimonious search is now invoked (i.e. the principle of Occam's razor is applied). That sub-set with the least number of biases is chosen first and the individual biases are quantified by performing least squares regression.

This result is then accepted if all the tank volume estimates and the product accountancy tank's plutonium inventory now agree with those measured. The result is only partially accepted if the bias estimates fail to correct the product accountancy tank residual.

5.2.2.3 Under-defined A matrix – non -unique solutions

Non-unique solutions exist if every tank in the tank-set is affected by a bias. To enable the estimation of these biases, a different method is required. Two criteria are introduced to enable a solution to be obtained:

1. There is no long-term build-up or loss of material within the tank-set.
2. The product accountancy tank's residual, $(\tilde{M}_{Pu} - \hat{M}_{Pu}) \approx 0$.

The first involves determining those relationships that are required to ensure zero build-up or loss over the long-term. If the (long-term) rate at which the estimated total volume of liquor contained in the tank-set ($\tilde{V}_{total}$) diverges from the actual total volume (V_{total}) is α, then:

$$\frac{d(V_{total} - \tilde{V}_{total})}{dt} = \alpha \tag{5.26}$$

where $\dfrac{dV_{total}}{dt} = \dfrac{1}{(t_f - t_s)}\left[\int_{t_s}^{t_f}\dfrac{dV_r}{dt}dt - \int_{t_s}^{t_f}\dfrac{dV_f}{dt}dt\right]$

and $\dfrac{d\tilde{V}_{total}}{dt} = \dfrac{1}{(t_f - t_s)}\left\{\left[\int_{t_s}^{t_f}\dfrac{dV_r}{dt}dt + \varepsilon_r\int_{t_s}^{t_f}\dfrac{dV_r}{dt}dt\right] - \left[\int_{t_s}^{t_f}\dfrac{dV_f}{dt}dt + \varepsilon_f\int_{t_s}^{t_f}\dfrac{dV_f}{dt}dt\right]\right\}$

hence:

$$\alpha = \varepsilon_f\int_{t_s}^{t_f}\frac{dV_f}{dt}dt - \varepsilon_r\int_{t_s}^{t_f}\frac{dV_r}{dt}dt = \varepsilon_f g_f - \varepsilon_r g_r \tag{5.27}$$

Thus if an estimate of α, i.e. $\tilde{\alpha}$, can be obtained then:

$$\tilde{\alpha} = \varepsilon_f g_f - \varepsilon_r g_r \tag{5.28}$$

which can be rearranged to provide an estimate for ε_r:

$$\varepsilon_r = \frac{\varepsilon_f g_f - \tilde{\alpha}}{g_r} \tag{5.29}$$

If the tank-set is of the conventional receipt/buffer/feed configuration, then ε_r can be substituted into Equation 5.19:

$$\begin{bmatrix} \delta_r \\ \delta_f \end{bmatrix} = AD_3 \begin{bmatrix} \varepsilon_r \\ \varepsilon_b \\ \varepsilon_f \end{bmatrix} = AD_3 \left\{ \begin{bmatrix} \frac{g_f}{g_r}\varepsilon_f \\ \varepsilon_b \\ \varepsilon_f \end{bmatrix} - \begin{bmatrix} \frac{\tilde{\alpha}}{g_r} \\ 0 \\ 0 \end{bmatrix} \right\} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} g_f\varepsilon_f \\ g_b\varepsilon_b \end{bmatrix} - \begin{bmatrix} \tilde{\alpha} \\ 0 \end{bmatrix} \tag{5.30}$$

which is singular if: $\varepsilon_b = \dfrac{\delta_f + g_f\varepsilon_f}{g_b}$ (5.31)

Therefore, using the first criterion reduces bias estimation for a three tank-set to an one-dimensional problem: if ε_f is hypothesised then the other two biases can be calculated provided that $\tilde{\alpha}$ is known. This parameter can be estimated by noting that any solution $\{\varepsilon_r^*, \varepsilon_b^*, \varepsilon_f^*\}$ must satisfy Equations 5.19, 5.28, and 5.31. If bias ε_f^* is chosen: $\varepsilon_f^* = 0 \Rightarrow \varepsilon_b^* = \dfrac{\delta_f}{g_b}$ and $\varepsilon_r^* = -\dfrac{\tilde{\alpha}}{g_r}$, then $\tilde{\alpha}$ can be estimated by performing linear regression (Equation 5.25) on the basis of one unknown parameter ε_r^*.

A unique solution can now be obtained by applying the second criterion. The second criterion is required because, as can be seen from $\tilde{\alpha} = \varepsilon_f g_f - \varepsilon_r g_r$, it is the scaled difference between the receipt and feed tank biases that determine the rate of divergence, and not their absolute values. Selecting biases that result in a zero α do not automatically lead to a correct long-term throughput of the tank-set. This can only be achieved by comparing the actual throughput, preferably through the tank-set itself, but more likely with the throughput as observed in the product accountancy tank. Thus ε_f is adjusted until:

$$\left|\hat{M}_{pu} - \tilde{M}_{pu}\right| < tol \tag{5.32}$$

where *tol* is likely to be relatively large to reflect the inaccuracy in the plutonium inventory estimate. Test cases are located in Chapter 6, Section 6.2.

It is important to understand that the analyses described can be performed regularly. Thus, for instance, a hypothesis can be made, which correlates with the day's data, then tested against data collected on subsequent days. It is difficult to 'pre-plan' a strategy for dealing with this eventuality until the actual data collection process begins.

5.3 Redistribution Tools

5.3.1 The Redistribution Tool

Due to difficulties in accurately estimating the volume transfers, the predicted plutonium distribution will always gradually diverge from the plant data. This drift will occur on a plant that is operating normally. To correct for this divergence, and also to provide an opportunity to detect medium-term disagreements, it is necessary to analyse the plant as a whole and then redistribute plutonium over time to minimise the divergence.

For each process unit in the plant (i.e. both tanks and process stages), the distribution tool allows the system to estimate the error between the simulated and real plant:

$$e_{unit} = [\tilde{P}u](\tilde{V} - \hat{V}) \quad (5.33)$$

The obvious solution is to correct the estimated transfers to minimise the error between the simulated and measured volume. However, as the estimated plutonium concentration is obtained from the distribution tool, it is necessary to re-run the tool if any of the corrected flow rates affect a process stage and hence the estimated plutonium concentration.

As the in-tank volume measurements have been analysed to estimate the transfers into and out of each tank, the transfer estimates will be biased if the volume measurements are biased. Therefore, the distribution model will also be affected by these biases and the predicted plutonium inventory in the product accountancy will diverge from that measured. Therefore it is important to identify and correct the volume transfers for the biases, for which a method is proposed.

The basic concept of the redistribution tool is simple. Material cannot cross the boundaries of the chemical separation areas, and so the tool seeks to move the excess material upstream or downstream until the errors are minimised. Each process unit has a tolerance from which material can be added or extracted e.g. if the tolerance on a tank was 50 grams, and its error was 34 grams, an extra 16 grams of plutonium could be redistributed into that tank or up to 34 grams could be extracted from that tank. In effect, the tool is attempting to level the error profile of the plant. As process stages can be viewed as hidden inventories, large amounts of material can be removed and placed into them.

For redistribution to take place, four assumptions have to be made:

1. The inventory of each process unit can only be in error by a specified amount, ∂_{unit} grams. ∂_{unit} should be selected on the basis of the standard deviations of both the volume and density measurements.
2. No material can be redistributed upstream of the first tank i.e. material flow measurements from the input accountancy vessel are assumed to be unbiased and error free for the time period in question.

3. No material can be redistributed downstream of the last tank i.e. material flow measurements from the output accountancy vessel are assumed to be unbiased and error free for the time period in question.
4. An estimate for the measurement bias on each tank is available and the distribution tool has been re-run to take these into account i.e. the Pu concentration is as correct as possible and the errors in the volume estimates is not due to biases on the measured signal.

Let ΔM_{unit} denote the incremental mass redistribution, i.e. the material balance moved between process units, and initially let $\Delta M_{unit} = 0$. Starting at the last process unit and ending in the first process unit in the plant:

1. Calculate the total error in the tank by summing the error in the tank with the material carried forward: $\bar{e}_{unit} = e_{unit} + \Delta M_{unit}$
2. Calculate the amount of material carried upstream to the next process unit: $\Delta M_{unit\cdot upstream} = \max(0, |\bar{e}_{unit}| - \partial_{unit}) * sign(\bar{e}_{unit})$
3. Revise the error in the unit as material has now moved upstream: $\bar{e}_{unit} = \bar{e}_{unit} - \Delta M_{unit\cdot upstream}$

If the first process unit is reached and $\Delta M_{unit} = 0$, then redistribution has been successful. Otherwise, the excess material can be redistributed downstream into any available process unit, and upon reaching the last process unit, $\Delta M_{unit} = 0$. In normal operation ∂_{unit} should be chosen to ensure this is the case.

For example, consider a four tank system with errors of 0, 10, 35, and 75 grams respectively and $\partial_{unit} = 50$ grams. Starting in Tank 4 with $\Delta M_4 = 0$:

1. $\bar{e}_4 = 75 + 0 = 75$
2. $\Delta M_3 = 75 - 50 = 25$
3. $\bar{e}_4 = 75 - 25 = 50$

Moving upstream to Tank 3:

4. $\bar{e}_3 = 35 + 25 = 60$

5. $\Delta M_2 = 60 - 50 = 10$

6. $\bar{e}_3 = 60 - 10 = 50$

Moving upstream to Tank 2:

7. $\bar{e}_2 = 10 + 10 = 20$

8. $\Delta M_1 = 20 - 50 = 0$

9. $\bar{e}_2 = 20 - 0 = 20$

Moving upstream to Tank 1:

10. $\bar{e}_1 = 0 + 0 = 0$

11. $\Delta M_1 = 0 - 0 = 0$

12. $\bar{e}_1 = 0 - 0 = 0$

Figure 5.3.1 shows the error profile before and after redistribution has taken place.

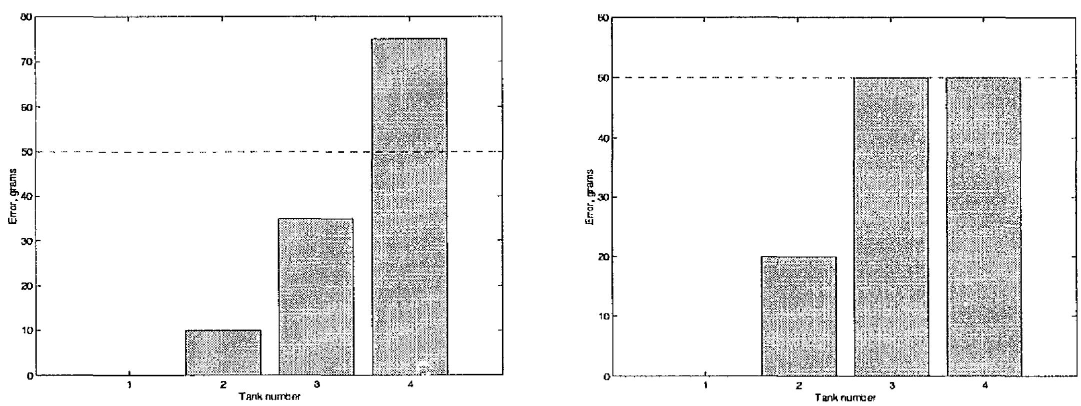


Figure 5.3.1 Error profile before (left) and after (right) redistribution. Tanks' error tolerance indicated by dashed line.

5.3.2 Medium-term Detection

Having set the tolerances on the process units to ensure redistribution on a normally operated plant, the occasion may occur when a process unit continues to diverge. Reasons for this may include a gradual diversion of material from a tank (i.e. a previously undetected low-magnitude flow that lasts for a period of days) or if the divergence is in the last tank only, a substitution of liquor with acid of similar density.

Identifying the process units where these diversions are taking place is not a straightforward task. If the diversion is in a buffer tank, then excess liquor builds up in the tank in the simulation, so the source of the diversion is easily identifiable. However, if the diversion is from a feeding or receiving tank, the excess plutonium will affect the process stage simulation, resulting in material propagating through the plant that is of incorrect concentration.

Since the simulated plant can be observed diverging from the real plant over a period of days, sampling on various tanks can be increased if a gradual diversion is suspected. The sample data can then be used to locate the process stage and its associated feeding and receiving tanks that are the source of the problem. Redistribution would then be applied on the basis that the problem is within the process stage.

The detection of an acid substitution would follow a similar procedure. The final tank in the plant would diverge, as the simulation would export excess quantities of plutonium when compared with the accountancy quantities. Again the build up would be observed and extra sampling would be undertaken. As before, the samples would aid in locating the process stage in whose vicinity the substitution had taken place.

In each case, appropriate flow rate correction terms would be generated and applied. An event would be created to describe the situation. Test cases are located in Chapter 6, section 6.3.

CHAPTER 6

TEST CASES

6.0 Introduction

This chapter illustrates how the system would respond in various situations. The additional safeguards system is designed to be effective in both the short and medium-term. The aim of this work was to examine the overall performance of the system without spending a large amount of time on detailed development. The simulation on which these results are based was crude, but unbiased. As the real plant data would undoubtedly be different, the tools were not optimised to their full extent. The results presented should be viewed with this understanding in mind.

The hypothetical reprocessing plant consisted of two solvent extraction cycles and a concentrator (Figure 6.0.1). The plant is thus constructed from 7 distinct segments.

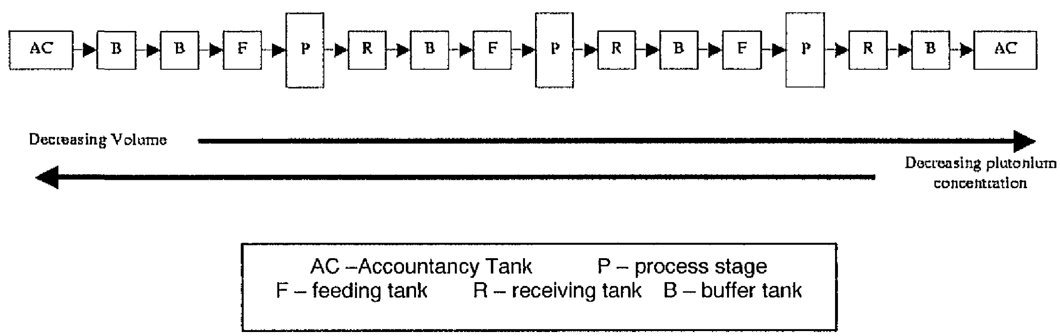


Figure 6.0.1 Arrangement of simulated plant.

The segments are (NB. The Pu concentrations are approximate):

1. Tank-set 1 consisting of four tanks after the input accountancy tank.
2. Cycle 1, Pu concentration increasing from 2g/l to 7g/l
3. Tank-set 2 consisting of three tanks.
4. Cycle 2, Pu concentration increasing from 7 g/l to 31.5 g/l
5. Tank-set 3 consisting of three tanks
6. Concentrator, Pu concentration increasing from 31.5 g/l to 220 g/l
7. Tank-set 4 consisting of two tanks, the second of which is connected to the product accountancy tank.

Figures 6.0.2 to 6.0.5 give the volume and density histories of each tank over the first 5000 minutes of a 4-day period.

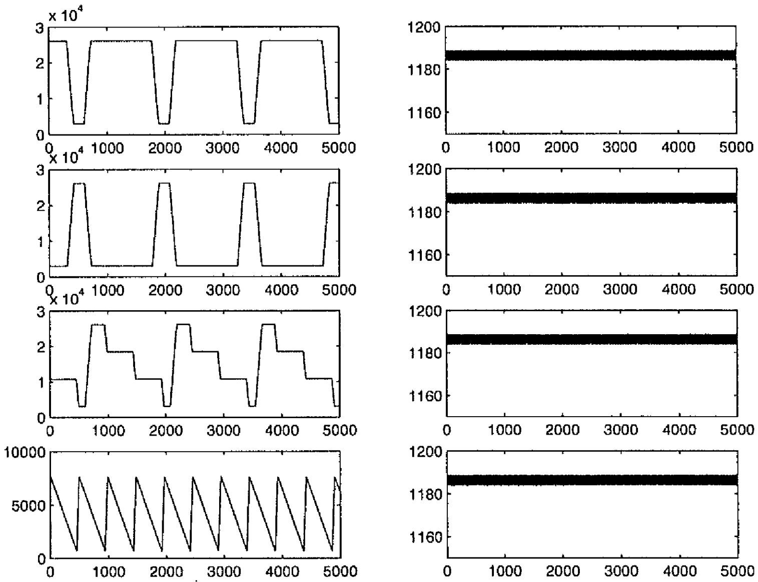


Figure 6.0.2 Tank-set 1, volume (litres) and density (g/l) plots

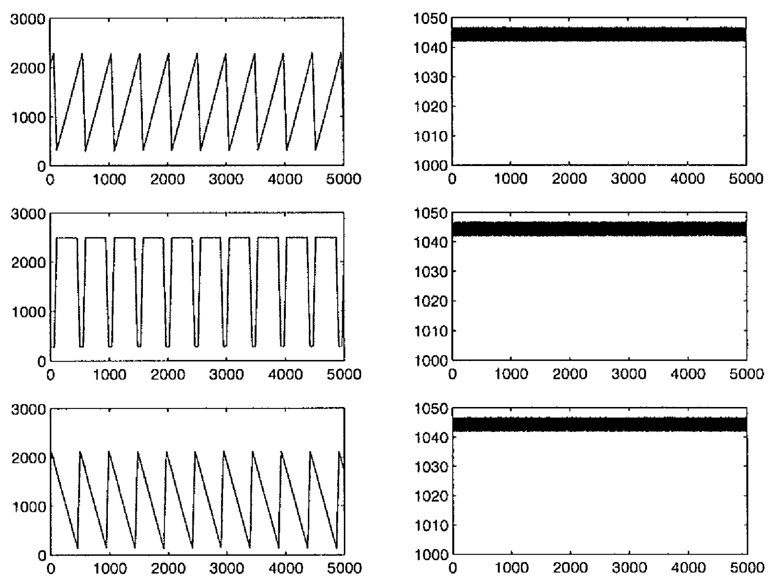


Figure 6.0.3 Tank-set 2, volume (litres) and density (g/l) plots

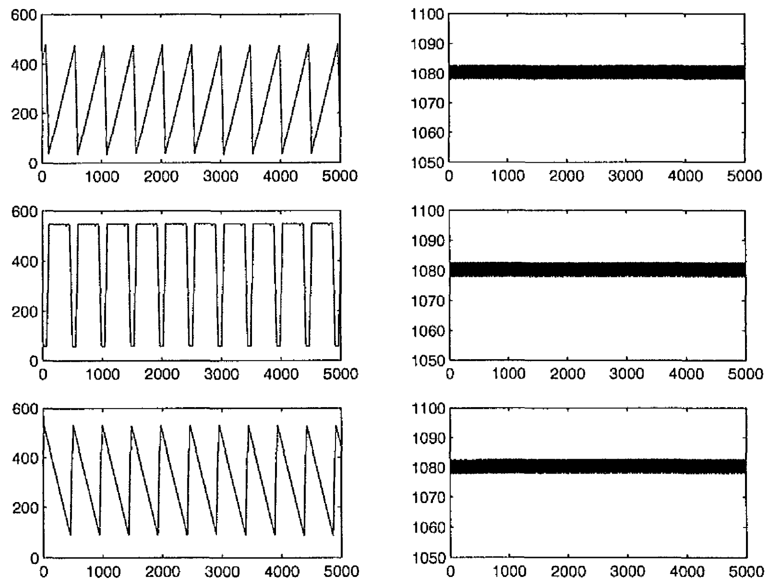


Figure 6.0.4 Tank-set 3, volume (litres) and density (g/l) plots

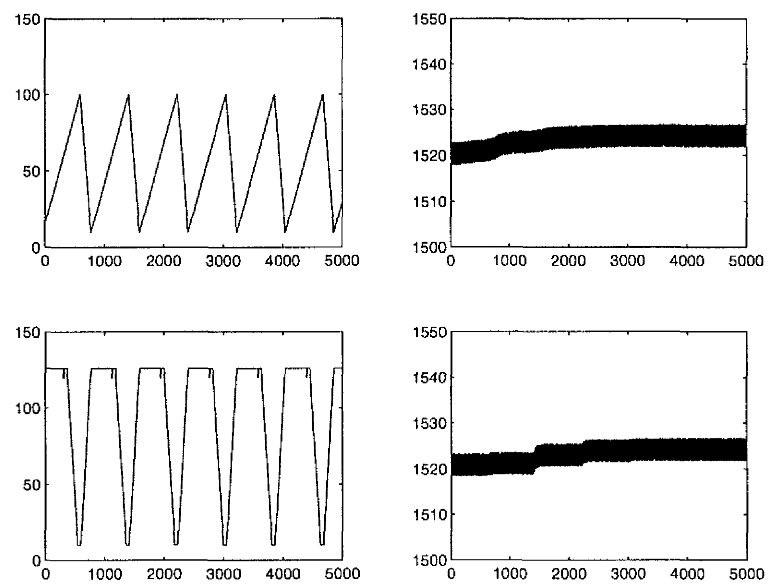


Figure 6.0.5 Tank-set 4, volume (litres) and density (g/l) plots

The simulation errors generated by comparing the simulation output with the plant data for the 'clean' plant (i.e. no disagreements) are shown in table 6.0.1 and the re-distributed errors (after 4 days) in table 6.0.2.

Process Unit	1 day	2 days	3 days	4 days
Tank 1	0.01	0.21	0.65	0.71
Tank 2	-9.60	-9.81	-9.98	-9.96
Tank 3	-0.04	-0.01	-0.37	-0.47
Tank 4	-6.51	-6.41	-8.33	-8.76
Solex 1	-0.01	-0.01	-0.02	-0.01
Tank 5	-1.19	0.39	-4.13	-3.46
Tank 6	1.50	0.73	0.26	-0.06
Tank 7	-7.26	-6.26	-4.84	-4.75
Solex 2	0.19	0.21	0.06	0.06
Tank 8	-38.99	-43.28	-75.09	-118.18
Tank 9	1.09	1.88	-5.75	-9.06
Tank 10	18.16	86.10	142.47	191.05
Concentrator	-0.75	-2.27	-1.24	3.14
Tank 11	1.56	10.80	18.04	22.60
Tank 12	-51.44	-51.64	-150.71	-61.99

Table 6.0.1: showing errors increasing over 4 days

Process Unit	Errors if no redistribution	Redistribution
Tank 1	0.71	0.71
Tank 2	-9.96	-9.96
Tank 3	-0.47	-0.47
Tank 4	-8.76	-8.76
Solex 1	-0.01	-0.01
Tank 5	-3.46	-3.46
Tank 6	-0.06	-0.06
Tank 7	-4.75	-4.75
Solex 2	0.06	0.06
Tank 8	-118.18	-36.19
Tank 9	-9.06	50.00
Tank 10	191.05	50.00
Concentrator	3.14	3.14
Tank 11	22.60	10.61
Tank 12	-61.99	-50.00

Table 6.0.2: redistribution if performed once at the end of 4 days

The test cases are split into three categories; each designed to test a different aspect of the system tools. These categories are:

1. Short-term disagreements (Section 6.1).
2. Gross biases (Section 6.2).
3. Medium-term disagreements (Section 6.3).

Within each section, a variety of different test cases are presented. It should be noted that the system is not optimised, and so the results may be somewhat clouded.

6.1 Short-term Test Cases

6.1.1 Abrupt Diversion from Cycle 2 Outlet.

Amount: 20 litres of liquor at Pu concentration of approximately 31.5 g/l.

Duration: 60 minutes.

Location: Cycle 2 outlet – molarity assumed to be constant.

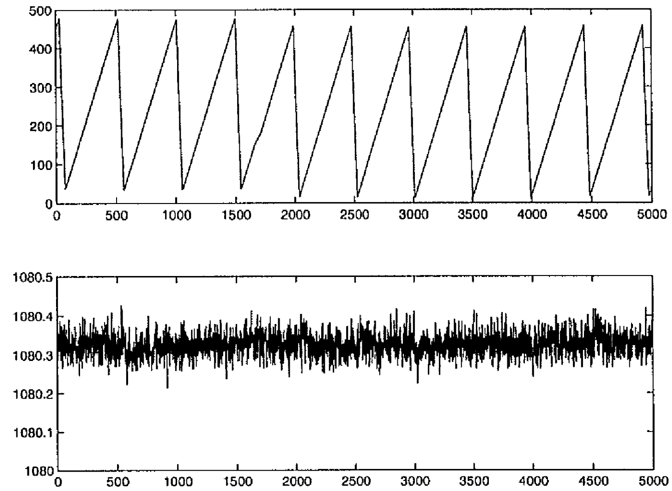
Effect: Temporary reduction in liquor entering receiving tank (tank 8).

This test case is designed to illustrate the response of the volume analysis tools described in Chapter 3 to an abrupt diversion of material during the filling of a receipt tank.

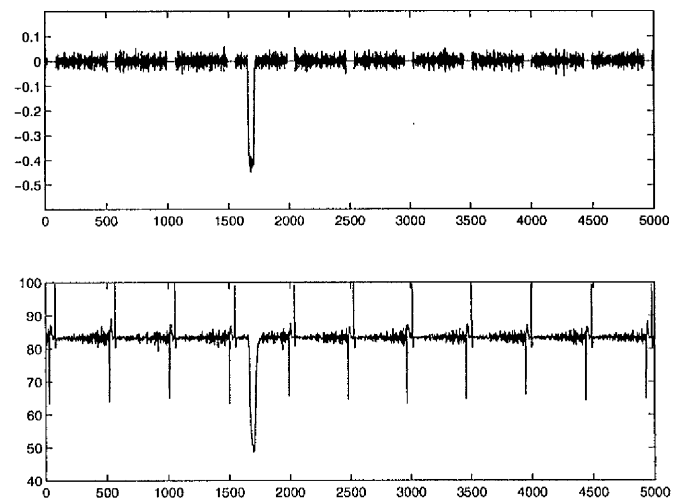
Figures 6.1.1.1 and 6.1.1.2 show the volume and density plots that are observed in tank 8: note the incident between 1500 and 2000 minutes in Figure 6.1.1.1. The outputs of the flow rate error observer (Section 3.3.2) and the X observer (Section 4.2.1) are shown in Figures 6.1.1.3 and 6.1.1.4. Note the irregularity from about 1700 minutes of duration 60 minutes. This is the first indication that material may be missing.

The irregularity is identified using a cusum-based detector (Section 3.4), the test signals of which are plotted in Figures 6.1.1.5-7. The 'start', 'stop' times and time history pertaining to the incident are then extracted for diagnosis. A sub-event is created to represent the irregularity.

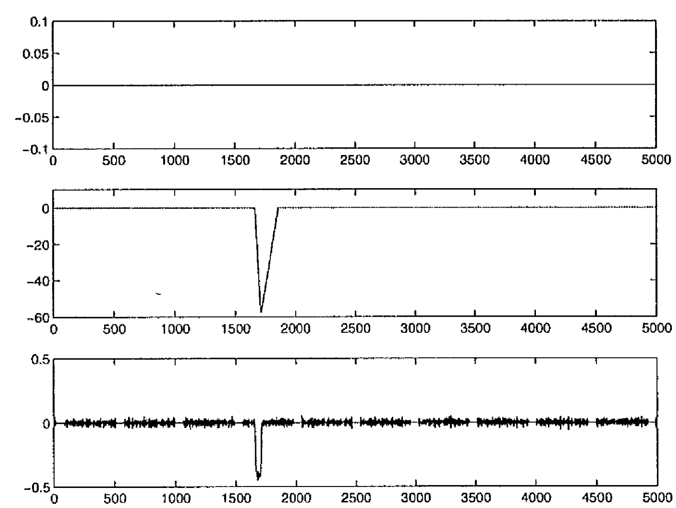
The three diagnostic tools (Section 3.5) are invoked concurrently: one examines the possibility that material may be transferred to hidden inventory (Figure 6.1.1.8); another examines the possibility of a simultaneous change in Pu concentration and acid molarity (Figures 6.1.1.9 & 6.1.1.10); the final possibility examined is a simultaneous change in unspecified and Pu concentration (Figures 6.1.1.11 and 6.1.1.12). The various possibilities would now be scored on their probability, and entered as possible diagnoses of the sub-event. The movement to hidden inventory is the most probable as it involves a single deviation and does not involve physically impossible concentrations of material.



Figures 6.1.1.1 & 6.1.1.2 Tank 8 volume (litres) and density (g/l)



Figures 6.1.1.3 & 6.1.1.4 Tank 8 observer outputs: flow rate error (l/min) and X (g/l)



Top: Positive test signal (detected if |signal| > some threshold)
Middle: Negative test signal (detected if |signal| > some threshold)
Bottom: Input signal

Figures 6.1.1.5-7 Tank 8 Detectors

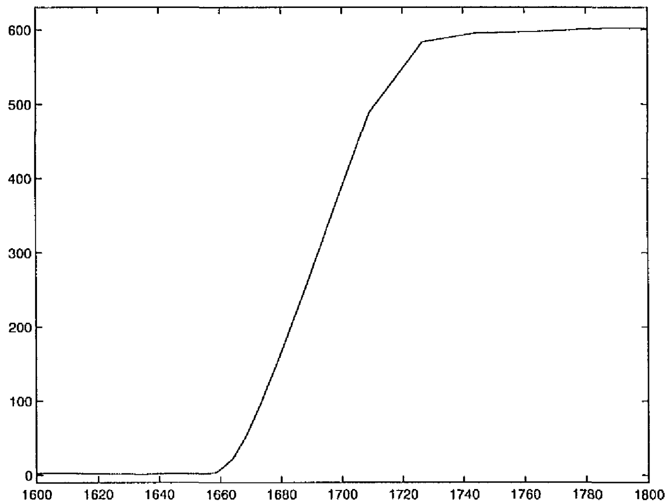
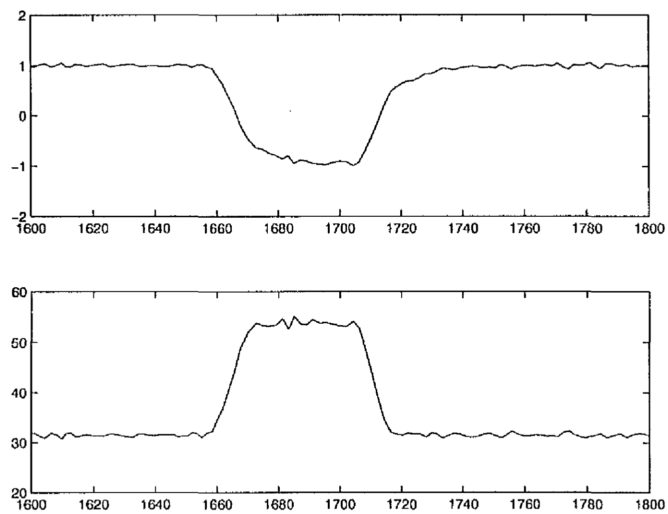


Figure 6.1.1.8 Hidden inventory diagnosis (grams of Pu)



Figures 6.1.1.9 & 6.1.1.10 Acid (mol/l) and Pu diagnosis (g/l)

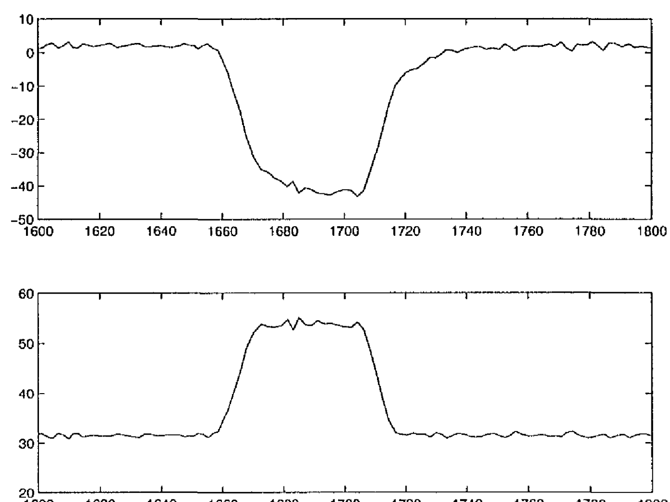


Figure 6.1.1.11 & 6.1.1.12 Unspecified (g/l) and Pu diagnosis (g/l)

6.1.2 Abrupt Diversion from Tank 8 during Export

Amount:	20 litres of liquor at Pu concentration of approximately 31.5 g/l.
Duration:	15 minutes, during batch transfer out.
Location:	Tank 8 to tank 9, batch transfer.
Effect:	Volume of liquor imported by tank 9 less than exported by tank 8.

This test case is designed to illustrate the response of the volume analysis tools described in Chapter 3 to an abrupt diversion of material during the emptying of a receipt tank.

This movement is characterised by an increase in the liquor remaining in tank 8 in the short-term simulation because the flow out is derived from the buffer tank (tank 9). Figure 6.1.2.1 shows the flow rate error observer output from tank8. There is no indication that material may be missing. The volume estimate from the short-term simulation is plotted in Figure 6.1.2.1 together with the actual tank volume measurements. Note the divergence between the two signals on the second batch transfer (940 minutes onwards).

Figures 6.1.2.3-5 shows the short-term simulation error signal that is input to the cusum-based detector (Section 3.4) and the two detector (Boolean) outputs. A 'start' time of about 920 minutes and a time history starting from 900 minutes are automatically extracted. A sub-event is created to represent the irregularity.

Up to four diagnostic tools (Section 3.5) are now invoked concurrently to diagnose the irregularity. The first looks at a transfer out to hidden inventory. The other three look at the possibility of additional material being added to the tank: plutonium nitrate solution of a similar concentration, plutonium nitrate solution of different concentration or acid. In each case a material movement would be obtained, but the addition of material is not physically meaningful and so these diagnoses have low probabilities. Thus the first diagnosis (Figure 6.1.2.6), transfer out to hidden inventory, is the most probable diagnosis of the sub-event.

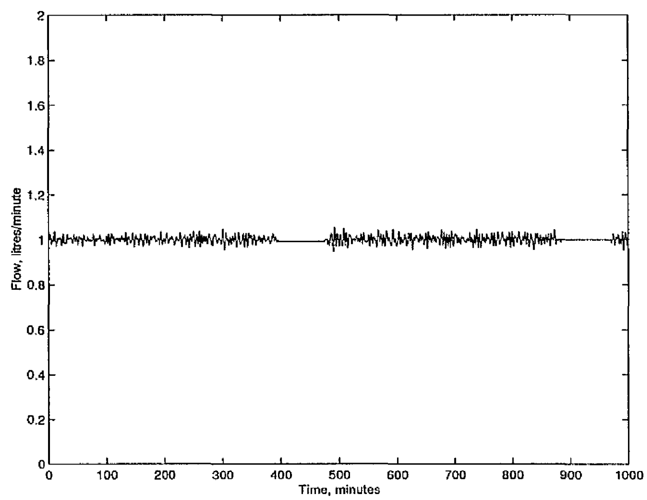


Figure 6.1.2.1 flow rate error observer signal for tank 8 (l/min)

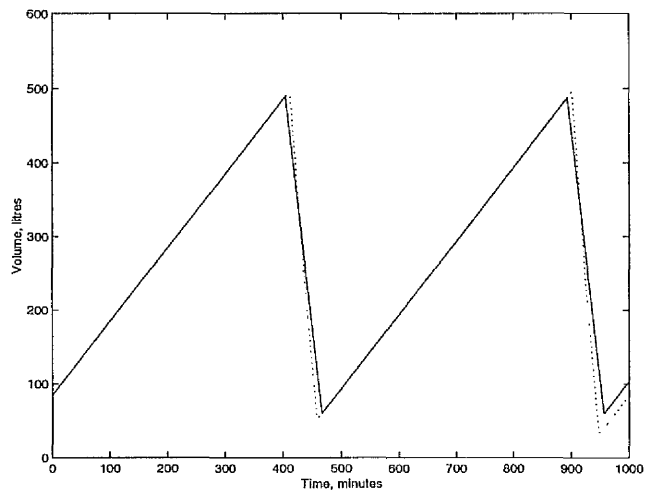
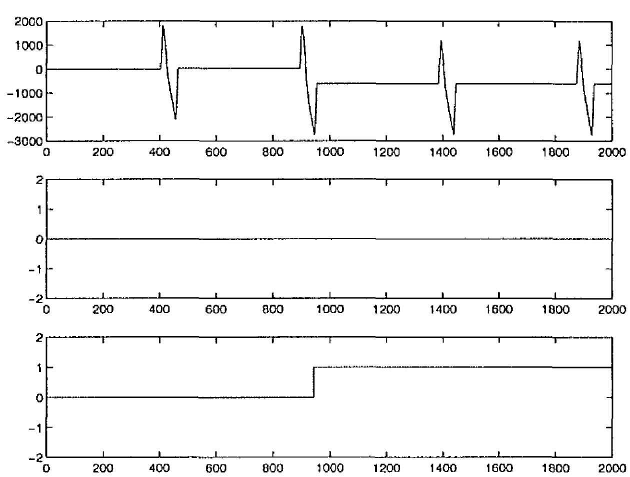


Figure 6.1.2.2 Tank 8 volume (litres). Prediction and measurements (dashed)



Top: Simulation error signal (litres)
Middle: Positive test signal (detected if |signal| = 1)
Bottom: Negative test signal (detected if |signal| =1)

Figure 6.1.2.3-5 Tank 8 detector signals

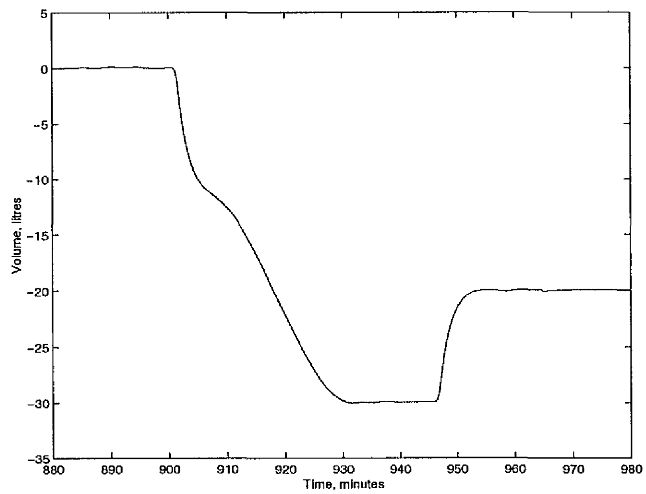


Figure 6.1.2.6 Hidden Inventory Diagnosis (grams of Pu)

6.1.3 Temporary Increase in Cycle 2 Inventory

Amount:	600 grams of Plutonium.
Duration:	12-hour period.
Location:	Cycle 2 inventory.
Effect:	Reduction in density of liquor exported to receiving tank (tank 8).

This test case is designed to illustrate the response of the density analysis tools described in Chapter 4 to a temporary increase in the inventory of the second solvent extraction plant. In this instance, 600 grams of plutonium 'disappears' into Cycle 2's inventory and then 'reappears' 12 hours later (Figure 6.1.3.1). This movement is characterised by transient decreases and increases in the density of liquor entering the receiving tank as the plutonium 'disappears' and 'reappears'. As the liquor consists of four components, it is difficult to attribute fluctuations in density with fluctuations in plutonium concentration. Density may also fluctuate due to variations in acid molarity or unspecified components concentration. As a consequence of this detection must be relatively insensitive and diagnosis must accommodate the various possibilities.

Figure 6.1.3.2 shows the output of the X-observer (Section 4.2.1). Note the pair of symmetrical irregularities, the first a negative irregularity, the second a positive irregularity. If these irregularities were identified by the cusum-based detector (Section 4.2.2) then the 'start' and 'stop' times and time history pertaining to the two irregularities would be extracted automatically. A sub-event is created to represent each irregularity.

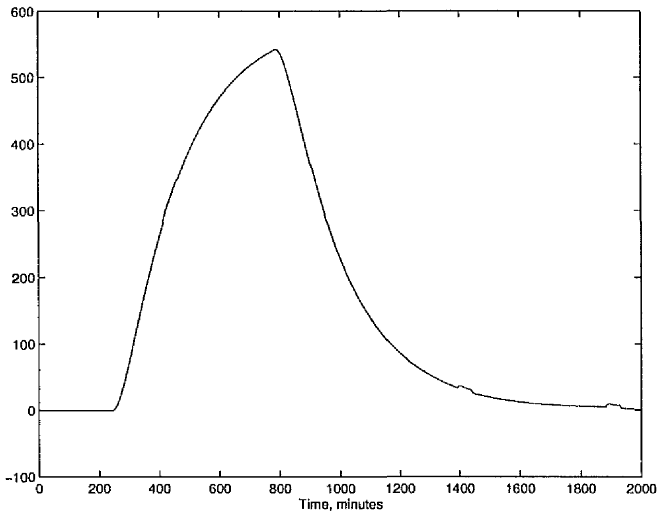


Figure 6.1.3.1 Transient change in Cycle 2 plutonium inventory (grams)

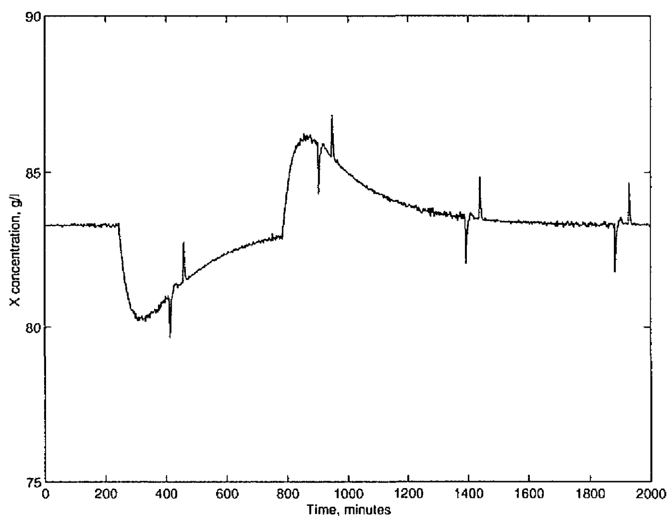


Figure 6.1.3.2 X-observer output for Cycle 2 receiving tank (g/l)

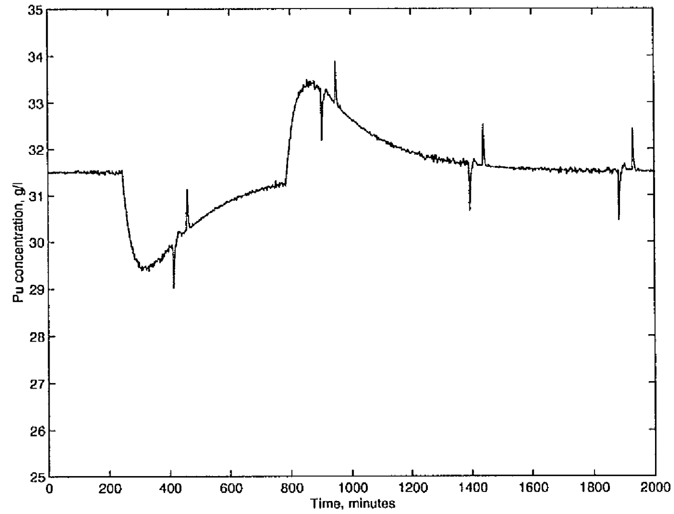


Figure 6.1.3.3 Plutonium concentration diagnosis (g/l)

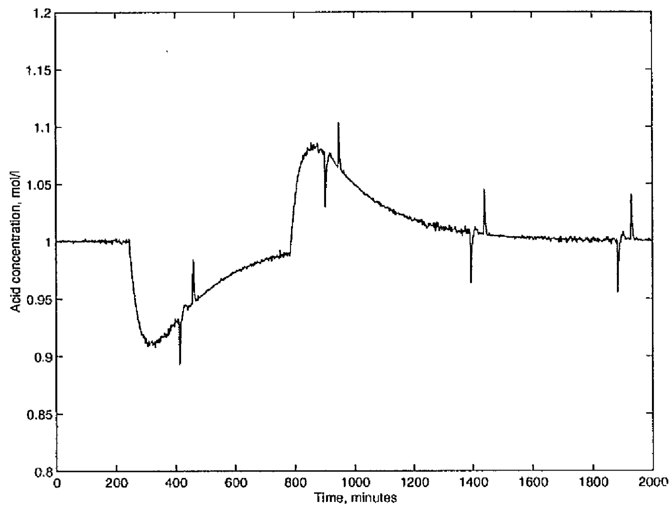


Figure 6.1.3.4 Acid molarity diagnosis (mol/l)

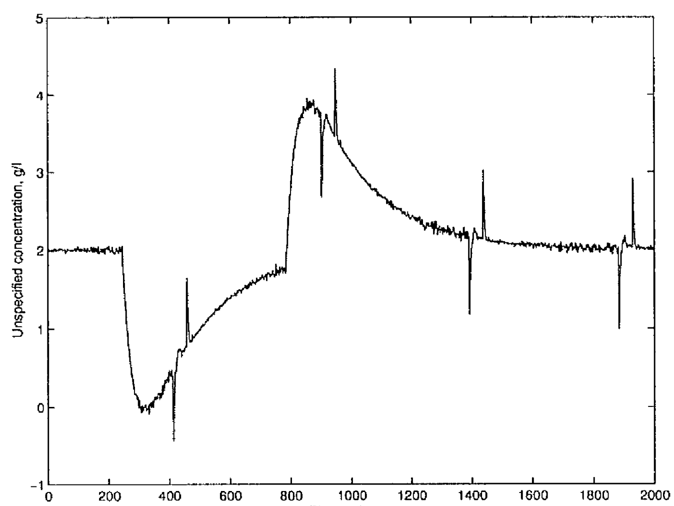


Figure 6.1.3.5 Unspecified concentration diagnosis (g/l)

The three diagnostic tools (Section 4.2.3) are now invoked concurrently. The first determines the change in Pu concentration that is required to explain the variation (Figure 6.1.3.3). The second the variation in acid molarity required (Figure 6.1.3.4) and the third the variation in the unspecified component's concentration (Figure 6.1.3.5) required to explain the variation. Probabilities and descriptions are then created for each possibility, the most likely explanation having the highest probability. These are now the diagnoses of the sub-event.

6.2 Gross Bias Test Cases

A number of test cases were examined to show the effect of multiplicative biases and how the various procedures described locate and estimate them. Table 6.2.1 indicates which tank volumes were used in the estimation of the various flow rates.

	Flow in	Flow out
Tank 1	Input Acc. Tank	Tank 2
Tank 2	Tank 2	Tank 2
Tank 3	Tank 2	Tank 3
Tank 4	Tank 3	Tank 4
Tank 5	Tank 5	Tank 6
Tank 6	Tank 6	Tank 6
Tank 7	Tank 6	Tank 7
Tank 8	Tank 8	Tank 9
Tank 9	Tank 9	Tank 9
Tank 10	Tank 9	Tank 10
Tank 11	Tank 11	Tank 11
Tank 12	Tank 11	Output Acc. Tank

Table 6.2.1: Identifies the tank volumes used to estimate the flow rates

It is assumed that the errors caused by sampling and evaporation are negligible. The test cases were designed to test the various procedures developed. In particular, the testing focussed upon the under-defined problem procedures. In each case the analysis was performed after 2 days to allow the errors to reach a sufficient level to exceed the tolerances. The value of the tolerances is a decision for the implementers of the system.

6.2.1 Over-defined and Square 'A' Matrix Test Case

As the procedure for estimating the biases when the matrix A is over-defined or square is straightforward, two test cases are presented. The first tank-set consists of a three buffer tanks and a feed tank, and so has a square A matrix (Table 6.2.1).

6.2.1.1 Single Tank Biased in Tank-set 1

The first, simple test case is a +1% measurement bias on Tank 3's volume. Note that since tank 3 is not a feed/receipt tank, the Pu error in Tank 12 is not included in the results. The resulting volume errors and the bias estimates obtained are shown in Table 6.2.2.

Tank	Error (litres)	Bias Estimate (%)
1	0.24	-
2	-4.64	0
3	-463.36	0.97
4	383.13	0

Table 6.2.2 Volume errors and bias estimates for +1% on Tank 3

These estimates were then used to produce revised volume throughput estimates for tanks 2 to 4. , then Tank 1's bias was estimated on the bias of Tank 2's bias since the flow into Tank 1 is unbiased(input from accountancy tank) and the flow out derived from Tank 2's volume measurements. The resulting volume errors and the bias estimates are shown in Table 6.2.3.

Tank	Error (litres)	Bias Estimate (%)
1	0.24	0
2	-4.64	0
3	-463.36	1.002
4	383.13	0

Table 6.2.3 Volume errors and revised bias estimates for +1% on Tank 3

Three sets of data of data were now analysed, collected every 750 minutes, to show the effect of analysing more than one set of data at a time (Table 6.2.4). Note that there is little difference to just waiting for 2 days. Finally the corrected disagreements and volume throughputs after 2600 minutes are given in Table 6.2.5. The biases estimated were found to eliminate the disagreements.

Tank	Iteration 1 (%)	Iteration 2 (%)	Iteration 3 (%)
1	0.0	0.0	0.0
2	0.0	0.0	0.0
3	1.003	1.003	1.002
4	0.0	0.0	0.0

Table 6.2.4: Bias estimates for first three 750 minutes iterations

	No Correction		+1.002 Correction	
Tank	Error (litres)	Throughput (litres)	Error (litres)	Throughput (litres)
1	-4.74	46315	-4.74	46315
2	-5.62	46320	-5.62	46320
3	-468.83	38975	-4.72	38600
4	381.77	40951	-4.85	40951

Table 6.2.5: Corrected throughput and disagreements after 2600.0 minutes

6.2.1.2 All Tanks Biased in Tank-set 1

In this more complicated test case, Tank 1 is subjected to a bias of +1%, Tank 2 to a bias of –1%, Tank 3 to a bias of +1%, and Tank 4 to a bias of –1%. Since the 'A' matrix is square, Tank 12's Pu disagreement has no bearing on the result. The resulting volume errors and the bias estimates obtained are shown in Table 6.2.6. The biases estimated were found to eliminate the disagreements (Table 6.2.7). Note that although errors still exist, these are below tolerance for the tank-set.

Tank	Error (litres)	Bias Estimate (%)
1	463.40	1.00
2	-4.75	-0.98
3	-926.51	1.03
4	769.43	-0.94

Table 6.2.6 Volume errors and bias estimates for all tanks biased test case

Tank	Bias Estimate (%)	Error (litres)
1	1.00	4.58
2	-0.98	-5.52
3	1.03	-1.19
4	-0.94	9.28

Table 6.2.7 Volume errors and bias estimates for all tanks biased test case

6.2.2 Under-defined A matrix test cases

As the procedure for estimating the biases when the 'A' matrix is under-defined is more complex, four test cases are presented. The single tank examples have unique solutions and the 'all' tanks example a non-unique solution. The location of all the test cases is the third tank-set (tanks 8, 9, & 10). As the biases in the feed and receive tanks (8 & 10) will affect the Pu disagreement in Tank 12, it is included in the results.

6.2.2.1 Single Tank Biased in Tank-set 3

Three test cases are presented for the testing of the procedure for estimating the biases when the matrix A is under-defined and the solution is unique. The test cases were:

a). +1% on Tank 8.

b). +1% on Tank 9.

c). +1% on Tank 10.

Since biases in a feed or receipt tank affect the process cycles, the Pu error in tank 12 is included in the error table (Table 6.2.8). The biases estimated by the procedure are shown in Table 6.2.9.

Tank	Case 'a'	Case 'b'	Case 'c'
8 (litres)	23.26	-26.53	-1.90
9 (litres)	0.04	0.12	0.04
10 (litres)	0.47	26.57	-23.27
12 (Pu gms)	-307.69	-	729.46

Table 6.2.8: Error table for unique solution test cases

Tank	Case 'a'	Case 'b'	Case 'c'
8 (litres)	0.94	0	0
9 (litres)	0	1.08	0
10 (litres)	0	0	0.90

Table 6.2.9: Bias estimates for unique solution test cases

For each test case, the bias estimate was found to eliminate both the volume and Pu disagreements.

6.2.2.2 All Tanks Biased in Tank-set 3

A single test case is presented for the testing of the procedure for estimating the biases when the matrix A is under-defined and the solutions are non-unique. The third tank-set is again the location for the test. The test case was:

+5% on Tank 8 -5% on Tank 9 -5% on Tank 10

The volume throughput and volume & Pu disagreements for the tank-set (including Tank 12) are shown in Table 6.2.10.

Tank	Throughput	Litres	Disagreement	Litres	Pu (gms)
8	d_r	2728	δ_r	242.78	7353.75
9	d_b	2328	δ_b	-0.27	-0.54
10	d_f	2468	δ_f	1.57	54.98
12	-	-	δ_{12}	-2.17	-7010.46

Table 6.2.10 Volume throughput and volume & Pu disagreements

Linear regression was now applied assuming $\tilde{\varepsilon}_f = 0$ & $\tilde{\varepsilon}_b = 6.7*10^{-4}$ to obtain $\tilde{\varepsilon}_r =$ 9.4% and hence $\tilde{\alpha} = -256.4$. (Table 6.2.11). Table 6.2.11 also contains the case (performed for interest) where the regression was repeated but with Equations 5.28 and 5.31 solved assuming $\tilde{\varepsilon}_r = 0$. The revised disagreements for the volumes are also tabulated in Table 6.2.11. Associated with these solutions are the revised Pu errors (Table 6.2.12).

Tank	Biases	Solution 1	Revised Disagreement	Solution 2	Revised Disagreement
8	$\tilde{\varepsilon}_r$	9.4	21.91	0	-41.99
9	$\tilde{\varepsilon}_b$	0	-0.27	-10.9	-0.30
10	$\tilde{\varepsilon}_f$	0	1.57	-10.4	16.33
12	-	-	-2.17	-	-21.7

Table 6.2.11: Revised volume disagreements for unique solutions

Tank	Biases	Solution 1	Revised Disagreement	Solution 2	Revised Disagreement
8	$\tilde{\varepsilon}_r$	9.4	777.04	0	-1204.75
9	$\tilde{\varepsilon}_b$	0	-0.58	-10.9	-0.08
10	$\tilde{\varepsilon}_f$	0	59.82	-10.4	497.63
12	-	-	-416.61	-	1132.71

Table 6.2.12: Revised Pu disagreements for unique solutions

As neither of the unique solutions gave satisfactory results, an iterative search (based on $\tilde{\varepsilon}_r$) was performed. The Pu disagreements for the first three iterations are shown in Table 6.3.13: the third iteration would be deemed to be acceptable because *tol* < 180 (i.e. Equation 5.32) and the sum of the disagreements squared is reduced. The process could then be repeated with revised volume throughputs that are based upon the third iteration (Table 6.2.14).

Tank	First Iteration		Second Iteration		Third Iteration	
8	5	-84.28	7.5	445.97	6.00	130.05
9	-5.09	-1.01	-2.16	-0.78	-3.92	-0.83
10	-4.86	247.08	-2.10	107.25	-3.76	188.86
12	-	259.68	-	-131.71	-	103.75

Table 6.2.13: Revised Pu disagreements for iterative search

	No Correction		With Correction	
Tank	Error (litres)	Throughput (litres)	Error (litres)	Throughput (litres)
8	242.78	2728	2.30	2573
9	-0.27	2328	-0.28	2423
10	1.57	2468	5.66	2565

Table 6.2.14: Corrected throughput and disagreements after 2600.0 minutes

6.2.3 Plant Analysis Strategy

This section is not a test case as such, but describes the application of the bias estimating procedures to the entire plant. The example referred to in this strategy is what might be considered the hardest test case: +1%, -1% on alternate tanks throughout the plant. The volume and Pu disagreements are shown in Table 6.2.15. It is important to understand that the analyses described can be performed regularly. Thus, for instance, a hypothesis can be made, which correlates with the day's data, then tested against data collected on subsequent days. It is difficult to 'pre-plan' a strategy for dealing with this eventuality until the actual data collection process begins.

Tank	Disagreement (litres)	Disagreement (Pu)
1	463.40	926.80
2	-4.75	-9.49
3	-926.51	-1853.02
4	769.43	1538.86
5	220.21	1508.97
6	-0.00	-0.08
7	-221.80	-1522.06
8	-50.92	-1748.30
9	0.30	9.65
10	51.33	1617.73
11	-0.03	-7.06
12	1.17	-850.62

Table 6.2.15: Volume and Pu disagreements for worst case scenario

The analysis of the first tank-set is straightforward as the 'A' matrix is square. Thus it is relatively easy to obtain estimates for the biases for this tank-set (Table 6.2.16). The volume and Pu disagreements are then re-estimated on the basis of these results. (Table 6.2.17).

Tank	Bias
1	1.00
2	-0.98
3	1.03
4	-0.94

Table 6.2.16 Bias Estimates for Tank-set 1

Tank	Disagreement (litres)	Disagreement (Pu)
1	4.58	19.12
2	-5.52	-9.59
3	-1.19	8.57
4	9.28	19.19
5	220.21	1523.44
6	-1.17	0.60
7	-221.99	-1536.52
8	-50.92	-1764.79
9	0.03	10.55
10	51.27	1632.58
11	-0.03	-7.12
12	1.15	-126.28

Table 6.2.17: Volume and Pu disagreements. Tank-set 1 solved

The analysis for tank-set 2 would be of a similar pattern to that performed for the non-unique test case. Looking solely at the measurements local to the tank-set, the preferred solution would be {0%, -2%, 0%}.Unfortunately, as biases downstream of the tank-set also affect the Pu error in Tank 12, this solution would not satisfy the condition on Tank 12. So the analysis would have to be continued using flow meters etc. to help in the estimation of biases. And so on for the other tank sets.

6.3 Medium-term Test Cases

The test cases presented in this section are designed to illustrate the response of the medium-term tools described in Chapter 5 in various scenarios, assuming that multiplicative biases either do not exist or have been estimated.

6.3.1 Gradual Diversion from Buffer Tank

Amount: 0.02 litre/min diversion of liquor at approximately 31.5 g/l

Duration: Continuous.

Location: Tank 9, buffer tank.

Effect: Build-up in error in Tank 9.

In this instance, liquor was continuously diverted from Tank 9 (the buffer tank after Cycle 2). Depending on the tolerances specified, a disagreement would arise in the redistribution tool after a few days (Table 6.3.1.1). The same tool would examine the redistribution of these errors and identify the problem in the correct buffer tank (Table 6.3.1.2). Appropriate flow rate correction terms would then be generated and applied. An event describing the disagreement would also be created.

Process Unit	1 day	2 days	3 days	4 days
Tank 1	0.17	0.25	0.50	0.48
Tank 2	-9.26	-9.10	-9.14	-9.22
Tank 3	-0.19	-0.09	-0.38	-0.36
Tank 4	-6.70	-6.62	-7.25	-7.55
Solex 1	-0.00	-0.00	-0.01	-0.00
Tank 5	0.47	1.44	-1.87	-0.93
Tank 6	-0.39	0.51	-0.31	-0.48
Tank 7	-8.60	-8.61	-7.94	-8.10
Solex 2	0.11	0.06	-0.02	-0.18
Tank 8	49.00	93.79	154.93	163.48
Tank 9	497.15	868.45	1442.33	1819.11
Tank 10	93.04	215.09	374.42	473.31
Concentrator	-0.15	0.01	-0.40	5.76
Tank 11	1.98	12.37	20.63	25.18
Tank 12	-47.84	-20.38	-85.78	-22.29

Table 6.3.1.1 Plutonium mass disagreements: simulated vs. measured, Tank 9

Process Unit	Errors if no redistribution	Redistribution
Tank 1	0.48	0.48
Tank 2	-9.22	-9.22
Tank 3	-0.36	-0.36
Tank 4	-7.55	-7.55
Solex 1	-0.00	-0.00
Tank 5	-0.93	-0.93
Tank 6	-0.48	4.72
Tank 7	-8.10	50.00
Solex 2	-0.18	50.00
Tank 8	163.48	50.00
Tank 9	1819.11	2242.24
Tank 10	473.31	50.00
Concentrator	5.76	5.76
Tank 11	25.18	25.18
Tank 12	-22.29	-22.29

Table 6.3.1.2 Tank 9 gradual diversion, redistribution

6.3.2 Gradual Diversion from Tank 8 Inlet

Amount: 0.02 litre/min diversion of liquor at approximately 31.5 g/l

Duration: Continuous.

Location: Cycle 2 outlet / Tank 8 inlet.

Effect: Over estimation of Pu throughput.

In this instance, liquor was continuously diverted from the Cycle 2 outlet (Tank 8 inlet). Depending on the tolerances specified, a disagreement would arise in the redistribution tool after a few days (Table 6.3.2.1). This build-up would trigger the collection of samples, which would then be compared with the simulation results (Table 6.3.2.2). This would indicate that the problem was located in the vicinity of Cycle 2. The redistribution tool would then redistribute the errors (table 6.3.2.3).

Appropriate flow rate correction terms would then be generated and applied. An event describing the disagreement would also be created.

Process Unit	1 day	2 days	3 days	4 days
Tank 1	0.24	0.30	0.52	0.72
Tank 2	-8.51	-8.61	-8.58	-8.52
Tank 3	-0.21	-0.23	-0.23	-0.54
Tank 4	-6.57	-6.65	-7.18	-7.39
Solex 1	-0.01	0.00	-0.01	-0.00
Tank 5	0.59	1.18	-1.91	-1.45
Tank 6	-1.00	0.22	-0.55	-0.55
Tank 7	-8.46	-7.87	-7.29	-8.05
Solex 2	0.02	-0.00	-0.06	-0.24
Tank 8	-27.38	-42.92	-109.76	-153.40
Tank 9	2.08	1.11	1.78	2.80
Tank 10	11.33	78.88	129.04	173.69
Concentrator	-1.70	-3.54	-1.35	5.09
Tank 11	11.45	6.33	19.98	18.75
Tank 12	130.08	512.59	1180.83	1852.60

Table 6.3.2.1 Plutonium mass disagreements: simulated vs. measured, Tank 8

Process Unit	Sample	Simulated
Tank 3	2.00	2.00
Tank 6	7.00	6.9992
Tank 9	31.45	31.98
Tank 12	221.78	226.61

Table 6.3.2.2 Plutonium sample data (g/l)

Process Unit	Errors if no redistribution	Redistribution
Tank 1	0.72	0.72
Tank 2	-8.52	-8.52
Tank 3	-0.54	-0.54
Tank 4	-7.39	-7.39
Solex 1	-0.00	-0.00
Tank 5	-1.45	-1.45
Tank 6	-0.55	-0.55
Tank 7	-8.05	-8.05
Solex 2	-0.24	1599.29
Tank 8	-153.40	50.00
Tank 9	2.80	50.00
Tank 10	173.69	50.00
Concentrator	5.09	50.00
Tank 11	18.75	50.00
Tank 12	1852.60	50.00

Table 6.3.2.3 Tank 8 inlet gradual diversion, redistribution on basis of sample data

6.3.3 Substitution of Solution with Acid

Amount:	50 litres of liquor at approximately 31.5 g/l
Duration:	short.
Location:	Tank 9.
Effect:	Over estimation of Pu throughput.

In this instance, a relatively large quantity of liquor, (50 litres at approximately 31.5 g/l of Pu) is replaced in the buffer tank upstream of the concentrator with a quantity of nitric acid of the same volume and density (50 litres of 2.32 molarity).

In theory, it is possible to perform the switch 'cleanly', i.e. no variation in the volume or density of the tank should be observed. In practice, however, the probability of achieving this must be very small, so some variation may be observed. There are numerous possibilities in the way in which the substitution may affect the volume and density measurements so here the worst case scenario is assumed: that any variation is indistinguishable from the process noise. The implication of this is that the detection and diagnosis is reliant upon corroborating evidence.

Although there would be various short-term effects on the units downstream, these would not be detectable, so detection is based on observing the medium term effects through the plutonium balance (redistribution tool). Table 6.3.3.1 shows the plutonium balance over the day of the incident and the following day. Note that the error in tank 12 takes time to build-up as the substituted solution propagates through the upstream units.

This build-up would trigger the collection of samples, which would be compared with the simulation results. Depending upon the timing, these may or may not provide evidence that the substitution has occurred. Either way, the problem would be diagnosed as being in the vicinity of the concentrator. The redistribution tool would then redistribute the errors.

Appropriate flow rate correction terms would then be generated and applied. An event describing the disagreement would also be created.

Process Unit	1 day	2 days
Tank 1	0.19	0.39
Tank 2	-9.40	-9.37
Tank 3	-0.04	-0.09
Tank 4	-3.94	-4.26
Solex 1	-0.01	-0.01
Tank 5	-0.04	2.71
Tank 6	-0.60	-0.19
Tank 7	-6.58	-7.75
Solex 2	-0.23	-0.33
Tank 8	-32.87	-50.86
Tank 9	3.71	0.22
Tank 10	-2.82	7.45
Concentrator	7.21	0.27
Tank 11	4.82	9.06
Tank 12	970.50	1539.33

Table 6.3.3.1 Acid substitution errors

Chapter 7

CONCLUSIONS AND FURTHER WORK

7.0 Material Safeguards

As pointed out by Burr and Wangen (1996b), conventional material accountancy methods will be unable to satisfy the protracted loss detection goal specified by the International Atomic Energy Agency when applied to large scale reprocessing plants. Therefore, there is a requirement for additional systems that enhance the conventional accountancy approach, and one possible system has been introduced in this thesis. The system proposed can fulfil many of the roles envisaged by Burr and Wangen (1996a), in assisting the safeguarding of nuclear material. In particular, the system has the ability to reconstruct plant transfers from minimum instrumentation and hence identify abnormal processes, such as diversions of material.

7.1 The Additional Safeguards System

It is appreciated that development of the additional safeguards system proposed in this thesis is curtailed somewhat by the lack of real data. The system is designed for confirming that a facility is operating as declared by the operator. The system itself is largely unproven, based on software developments that would evolve as the plant is commissioned.

An extensive programme of work would be required to ensure a successful implementation on a commercial facility, much thought has gone into ensuring that, having invested so much in the implementation, the evaluation tools can be evolved so that the system performs as expected. It must be emphasised that the system's success depends upon the quality of the data collection system. It is crucial that appropriate data is collected; although the analytical tools can evolve, the data cannot i.e. one cannot go back and collect the same data again.

The various case studies have demonstrated that the implementation of such a system would provide additional assurances that the plant is operating as declared. The focus of the case studies has been on how the system would work, rather than on measuring the system's ability to detect and isolate a particular incident. This is because quantitative results would be misleading unless they were based on data that resembled real data, which is not currently available.

7.2 Summary

It has been demonstrated that the proposed additional system has the potential to be a valuable aid for inspectors responsible for nuclear materials safeguards, providing additional short-term and medium-term assurances to supplement traditional materials accountancy methods. Working prototypes of the tools described within this thesis have been produced and evaluated using data sets pertaining to a hypothetical plant. Features of the system described within this thesis, which are likely to be of particular use, are:

- simple, practical algorithms;
- transfer estimation;
- ability of the design to evolve to meet future requirements;
- methods used can be applied to information poor plants, where traditional methods have problems;
- the potential to detect and diagnose short-term disagreements;
- the potential to detect and diagnose medium-term disagreements;
- the potential to estimate plutonium distribution within the chemical process area at any point in time;
- a method to detect and identify gross biases has been established.

7.3 Component Summary

It is somewhat difficult to draw conclusions about the tools without having the opportunity of testing them on real data.

The volume analysis tools described in Chapter 3 are robust, simple designs that are easily adapted if need be. As several of their constituent components have already been shown to perform adequately with real data (Appendix 1), the application of the tools to real data should be straightforward. Since the Modified Shewhart Control

Chart is easier to set-up and tune than the V-mask, it is the preferred change detection algorithm. To calculate the flow rates, simulated annealing should only be used as a last resort. Although it guarantees the correct solution, the computational efficiency of the algorithm is low.

Observers are widely used throughout the system, not only in the volume analysis tools (Chapter 3) but also in the density analysis tools described in Chapter 4. They have been shown to be appropriate for the generation of detector signals and for the diagnosis of any anomalies that may be found.

The method developed for the estimation of gross multiplicative biases, presented in Chapter 5, has been shown to work on typical plant data where a limited number of tanks have volume measurements that are biased grossly. The case where all tank volume measurement systems have gross biases is considerably more difficult; although the method would give possible solutions, additional plant data would be needed to establish the validity of the estimates thus obtained.

The redistribution and the medium-term assurance tools perform as expected. The largest difficulty associated with the medium-term assurance tools is the decision as to when to redistribute and when to estimate gross biases. A solution to this problem is proposed, but this would probably require modification for use on a real plant.

7.4 Recommendations for Future Work

The work presented in this thesis can be considered to be a set of development guidelines for the future implementation of an additional safeguards system. There are many possible improvements and refinements that could be made to the system in the future, some possibilities include:

- development of a complete prototype system. This depends upon the availability of real plant data and/or improved simulated data;
- expand the system to cover design information verification. The system will provide an operational template for a plant once implemented, and this information can be used to detect unauthorised alterations to the plant design or operation;
- further research in data compression and warehousing.

REFERENCES

Albuquerque, J.S., and Biegler, L.T., (1996). Data Reconciliation and Gross-Error Detection for Dynamic Systems. *American Institute of Chemical Engineers Journal,* Vol. 42 (10), pp. 2841-2856.

Bagajewicz, M.J., and Jiang, Q., (1998). Gross Error Modelling and Detection in Plant Linear Dynamic Reconciliation. *Computers and Chemical Engineering.* Vol. 22 (12), pp. 1789-1809.

Barkema, G.T. and Newman, M.E.J. (1999). Monte Carlo Methods in Statistical Physics. *Clarendon Press,* Oxford, UK.

Barnard, G.A. (1959). Control Charts and Stochastic Processes. *Journal of the Royal Statistical Society (B),* Vol. 21, pp. 239-257.

Basseville, M. (1988). Detecting Changes in Signals and Systems. *Automatica,* Vol. 24 (3), pp. 309-326.

Basseville, M., and Nikiforov, I.G. (1993). Detection of Abrupt Changes: Theory and Application. *Prentice Hall Information and System Sciences Series, Prentice Hall Inc.*. New Jersey, USA.

Benedict, M., Pigford, T.H., and Levi, H.W. (1981). Nuclear Chemical Engineering, *McGraw-Hill series in Nuclear Engineering, 2nd Edition,* McGraw-Hill, USA.

Beyerlein, A. L., and Geldard, J.F. (1989). Contactor Inventory Variations and Nuclear Material Accounting for Reprocessing Systems. *Journal of Nuclear Materials Management*, Vol. 18, pp. 985-991.

BNFL (1996). Specification, Selection, Installation and Maintenance of Solution Mass Measurement, Issue 2. *UK Safeguards R&D Programme,* UKAEA Report SRDP-R227.

Bristol, E.H., (1990). Swinging Door Trending: Adaptive Trend Recording?. *Advances in Instrumentation,* Vol. 45 (2), pp. 749-754.

Burr, T. and Wangen, L. (1996a). Development and Evaluation of Methods for Safeguards Use of Solution Monitoring Data, *Los Alamos National Laboratory,* LA-13185-MS.

Burr, T. and Wangen, L. (1996b). Enhanced Safeguards Via Solution Monitoring, *Los Alamos National Laboratory,* LA-13186-MS.

Candy, J.V., and Rozsa, R.B. (1980). Safeguards Design for a Plutonium Concentrator – An Applied Estimation Approach, *Automatica,* 16, pp. 615-622.

Chatfield, C. (1996). Statistics for Technology: A Course in Applied Statistics, third edition (revised). *Chapman and Hall*, London.

Chen, W., (2000). University of Glasgow. Personal communication.

Cheung, J.T.Y, and Stephanopoulos, G., (1990a). On the Detection and Representation of Trends. *Advances in Instrumentation*, Vol. 45 (2), pp. 755-774.

Cheung, J.T.Y, and Stephanopoulos, G., (1990b). Representation of Process Trends, Parts I and II. *Computers and Chemical Engineering*, Vol. 14 (4-5), pp. 495-539.

Clarke, D.W., and Henry, M.P. (1993). The Self-Validating Sensor: Rationale, Definitions and Examples. *Control Engineering Practice*, Vol. 1, pp 585-610.

Cobb, D.D., Dayem, H.A., Baker, A.L., Ellis, J.H., Ehinger, M.H., Crawford, J.M., (1981a). Demonstration of Near-Real-Time Accounting: The AGNS 1980 Mini-runs. *Journal of Nuclear Materials Management*, Vol. 10, pp. 201-212.

Cobb, D.D., Dayem, H.A., Hakkila, E.A., Shipley, J.P., Baker, A.L., (1981b). Development and Demonstration of Near-Real-Time Accounting Systems for Reprocessing Plants. *Journal of Nuclear Materials Management*, Vol. 10, pp. 213-219.

Crowder, S.V. (1987). A Simple Method for Studying the Run-Length Distributions of Exponentially Weighted Moving Average Charts. *Technometrics,* Vol 29, pp. 152-161.

Crowe, C.M., Garcia Campos, Y. A., Hrymak, A., (1983). Reconciliation of Process Flow Rates by Matrix Projection, Part I: Linear Case. *American Institute of Chemical Engineers Journal,* Vol. 29, pp. 881-888.

Dantzig, G.B. (1963) (reprinted 1993). Linear Programming and Extensions. *Princeton Landmarks in Mathematics and Physics, Princeton University Press,* Princeton, New Jersey, USA.

De Kleer, J., and Brown, J.S. (1984). A Qualitative Physics based on Confluences. *Artificial Intelligence*, 24, pp. 7-83.

Ehinger, M. H. (1989). Process Monitoring in International Safeguards for Reprocessing Plants – A Demonstration. *Oak Ridge National Laboratory report,* ORNL/TM-10912, ISPO-253.

Ehinger, M.H., Kerr, H.T., Wachter, J.W., and Hebble, T.L. (1981). Process Monitoring for Reprocessing Plants Safeguards – A Summary Review. *Oak Ridge National Laboratory report,* ORNL/TM-10151, ISPO-255.

Fishman, G.S. (1996). Monte Carlo, Concepts, Algorithms, and Applications. *Springer Series in Operations Research, Springer-Verlag,* New York, USA.

Forbus, K.D. (1984). Qualitative Process Theory. *Artificial Intelligence*, Vol. 24, pp. 85-168.

Frank, P. (1990). Fault Diagnosis in Dynamic Systems Using Analytical and Knowledge-Based Redundancy – a Survey and Some New Results. *Automatica,* Vol. 26 (3), pp. 459-474.

Frank, P. (1994). Application of Fuzzy Logic to Process Supervision and Fault Diagnosis. *Safeprocess '94, IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes,* Vol. 2. Espoo, Finland.

Friedland, B. (1987). Control System Design: An Introduction to State-Space Methods. *McGraw-Hill Series in Electrical Engineering, McGraw-Hill Inc.,* USA.

Geiger,G., Werner, T., Matko, D. (2001). Knowledge-based Leak Monitoring for Pipelines. *IFAC Workshop on On-line Fault Detection and Supervision in the Chemical Process Industries*. Jejudo Island, Korea.

Gentil, S. and Montmain, J. (2000). Dynamic Causal Model Diagnostic Reasoning for Online Technical Process Supervision. *Automatica,* Vol. 36 (8), pp. 1137-1152.

Gertler, J., and Singer, D. (1990). A New Structural Framework for Parity Equation-based Failure Detection and Isolation. *Automatica,* Vol. 26 (2), pp. 381-388.

Gertler, J. (1991). Analytical Redundancy Methods in Fault Detection and Isolation. *Proceedings of IFAC/IMACS Symposium , Safeprocess'91,* Baden-Baden.

Graup V. (1972). Identification of Systems. *Van Nostrum.*

Hale, J.C., and Sellars, H.L., (1981). Historical Data Recording for Process Computers. *Chemical Engineering Progress*, November, pp. 38-43.

The Health and Safety Executive (1995). Thermal Oxide Reprocessing Plant (THORP) – The Regulation of THORP by HM Nuclear Installations Inspectorate', *HSE Books,* Sudbury, Suffolk, ISBN 0717610047.

Hess, R.A., Gao, C., and Wang, S.H. (1991). Generalized Technique for Inverse Simulation Applied to Aircraft Maneuvers. *Journal of Guidance*, Vol. 14 (5), pp. 920-926.

Hisa, T.C. (1977). System Identification – Least Squares Methods. *Lexington Books, DC Heath and Company*, Lexington.

Howell, J. (1994). Model-based Fault Detection in Information Poor Plants. *Automatica*, Vol. 30 (6), pp. 929-943.

Howell, J., and Scothern, S.J. (1995). Model-Based Diagnosis as an Aid to Anomaly Resolution, *UK Safeguards R&D Programme*, UKAEA Report SRDP-R226

Howell, J., and Scothern, S.J. (1997). A Physical-Model-Based Diagnostic Aid for Safeguarding Nuclear Material in a Liquor Storage Facility. *Journal of Nuclear Materials Management*, Vol. 25 (4), pp. 20-29.

Howell, J. and Miller, E.C., (2001a). Evaluation of Process Information to Obtain Additional Safeguards Assurances in Reprocessing Plants. *UK Safeguards R&D Programme*, UKAEA Report SRDP-R279.

Howell, J. and Miller, E.C., (2001b). Tailoring the Glasgow University Diagnostic Aid for the Product Storage Facility at TRP. *UK Safeguards R&D Programme*, UKAEA Report SRDP-R280.

IAEA (1992). The Structure and Content of Agreements Between the Agency and States Required in Connection with the Treaty on the Non-Proliferation of Nuclear Weapons, *International Atomic Energy Agency,* Vienna, INFCIRC/153 (Corrected).

IAEA (1999a). Training Course on Reprocessing Safeguards Instrumentation. *Section for Safeguards Training, Department of Safeguards.* Technical note, BNFL Sellafield, United Kingdom.

IAEA (1999b). JNFL Project: User Requirements and Functional Specifications for the Solution Monitoring System at the Rokkasho Reprocessing Plant. *International Atomic Energy Agency,* Vienna.

Isermann, R. (1984). Process Fault Detection based on Modelling and Estimation Methods – A Survey. *Automatica,* Vol. 20 (4), pp. 387-404.

Islam, B.M.N., Johnson, S.J., Sellinschegg, W.D. (1993). Meeting Timeliness Requirements in Reprocessing Plants, *Journal of Nuclear Materials Management,* October, pp. 19-24.

Johnson, N.L. (1961). A Simple Theoretical Approach to Cumulative Sum Control Charts. *Journal of the American Statistical Association*, Vol. 61, pp. 835-840.

Kalman, R.E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, Vol. 82, pp. 35-45.

Kalman, R.E. and Bucy, R.S. (1961). New Results in Linear Filtering and Prediction Theory. *Journal of Basic Engineering*, March, pp. 95-108.

Landat, D., Hunt, B.A., Koehne, W., Franssen, F., and Hope, D., (1997a). Portable Measurement Equipment for Unattended Process Monitoring and Verification of liquids in Tanks. *Proceedings of the 38th Annual meeting of the Institute of Nuclear Materials Management,* Phoenix, Arizona.

Landat, D., Hunt, B.A., Barratti, G., and Galli, M., (1997b). Development and Application of Portable Inspection Monitoring Systems for Tank Calibrations and Unattended Verification of Volume Measurements. *Proceedings of the 19th ESARDA Annual Symposium on Safeguards and Nuclear Material Management,* Montpellier, France.

Landat, D., Caviglia, M., and Hunt, B.A. (1998). Characterization and use of the Volume Long-Term Monitoring Device, VLTM. *Technical note N.I.98.92.* JRC Ispra

The LASCAR Forum (1992). Report of the LASCAR Forum: Large Scale Reprocessing Plant Safeguards, *International Atomic Energy Agency,* Vienna, STI/PUB/992.

Liebman, M. J., Edgar, T.F., Lasdon, L.S., (1992). Efficient Data Reconciliation and Estimation for Dynamic Processes using Nonlinear Programming Techniques. *Computers and Chemical Engineering*, Vol. 16 (10-11), pp. 963-986.

Lovett, J., Ikawa, K., Shipley, J., and Sellinschegg, D., (1982). Near-Real-Time Materials Accountancy, A Technical Status Report. *International Atomic Energy Agency,* IAEA-SM-260/145.

Luenberger, D.G. (1966). Observers for Multivariable Systems. *IEEE Transactions on Automatic Control*, Vol. AC-11, (2), pp. 190-197.

Luenberger, D.G. (1971). An Introduction to Observers. *IEEE Transactions on Automatic Control*, Vol. AC-16, (6), pp. 596-602.

Mah, R.S.H, Stanley, G., Downing, D., (1976). Reconciliation and Rectification of Process Flow and Inventory Data. *Industrial Engineering Process Design Development*, Vol. 15 (1), pp. 175-183.

Mah, R.S.H, and Tamhane, A.C., (1982). Detection of Gross Errors in Process Data. *American Institute of Chemical Engineers Journal,* Vol. 28, pp.828-830.

The Mathworks Inc.. Matlab and Simulink – registered products of The Mathworks Inc.,24 Prime Park Way, Natick, MA, USA.

Montgomery, D.C. (1996). Introduction to Statistical Quality Control, third edition. *Wiley*, Chichester.

Murray-Smith, D.J. (2000). The Inverse Simulation Approach: A Focussed Review of Methods and Applications, *Third Mathmod Symposium*, Vienna.

Narasimhan, S., and Mah, R.S.H., (1987). Generalised Likelihood Ratio Methods for Gross Error Identification. *American Institute of Chemical Engineers Journal,* Vol. 33, pp. 1514-1521.

Nelder, J.A. and Mead, R. (1965). A Simplex Method for Function Minimisation. *Computer Journal*, Vol 7, pp. 308-313.

Page, E.S. (1954). Continuous Inspection Schemes. *Biometrics,* Vol 41, pp. 100-114.

Patten, B.C. (1970). The Inverse Modelling Problem. *Simulation*, December, pp. 264-268.

Patton, R., Frank, P., and Clark, R. (1989). Fault Diagnosis in Dynamic Systems: Theory and Application. *Prentice Hall International Series in Systems and Control Engineering, Prentice Hall International,* London.

Powell, M.J.D. (1965). An Efficient Method for Finding the Minimum of a Function of Several Variables without Calculating Derivatives. *Computer Journal*, Vol 7, pp. 303-307.

Powell, M.J.D. (1968). A FORTRAN Subroutine for Solving Systems of Non-Linear Algebraic Equations. *United Kingdom Atomic Energy Authority,* AERE-R 5947.

Reilly, P., and Carpani, R., (1963). Application of Statistical Theory of Adjustment to Material Balances. *Proceedings of the 13th Canadian Chemical Engineering Conference.*

Roberts, S.W. (1959). Control Chart Tests Based on Geometric Moving Averages. *Technometrics*, Vol. 1 (3), pp. 239-250.

Rugh, W.J., (1996). Linear System Theory, Second Edition. *Prentice Hall Information and System Sciences Series, Prentice Hall Inc.*. New Jersey, USA.

Sanchez, M., Romagnoli, J., Jiang, Q., Bagajewicz, M., (1999). Simultaneous Estimation of Biases and Leaks in Process Plants. *Computers and Chemical Engineering*, Vol. 23 (7), pp. 841-857.

Shewhart, W.A. (1931). Economic Control of Quality of Manufactured Product. *MacMillan*, New York.

Sistu, P.B., Gopinath, R.S., Bequette, B.W., (1993). Computation Issues in Nonlinear Predictive Control. *Computers and Chemical Engineering*, Vol. 17 (4), pp. 361-366.

Soderstrom, T., and Stoica, P. (1989). System Identification. Prentice Hall International Series in Systems and Control Engineering. *Prentice Hall International Ltd.*, London.

Thomson, D.G. and Bradley, R. (1990). Development and Verification of an Algorithm for Helicopter Inverse Simulation. *Vertica*, Vol. 14 (2), pp. 185-200.

Tsutsumi, M., Suyama, N., Kinuhata, T., Fukuda, G., and Koizumi, T., (1982). Result of Material Accountancy and Control and an Evaluation for MUF at the Tokai Reprocessing Plant. *International Atomic Energy Agency,* IAEA-SM-260/118

Van Ginneken, L.P.P.P. and Otten, R.H.J.M. (1989). The Annealing Algorithm. *Kluwer Academic Publishers Group*, Dordrecht.

Van Laarhoven, P.J.M. (1988). Theoretical and Computational Aspects of Simulated Annealing. *CWI Tract 51, Centrum voor Wiskunde en Informatica*, Amsterdam.

Venkatasubramanian, V. (2001). Process Fault Detection and Diagnosis: Past, Present and Future. *IFAC Workshop on On-line Fault Detection and Supervision in the Chemical Process Industries*. Jejudo Island, Korea.

Viswanadham, N. and Srichander, R. (1987). Fault Detection Using Unknown-Input Observers. *Control-Theory and Advanced Technology,* Vol. 3 (2), pp. 91-101.

Viswanadham, N. and Minto, K.D. (1988). Robust Observer Design with Application to Fault Detection. *Proceedings of the American Control Conference 1988*, pp. 1394-1398.

Willsky, A.S., (1976). A Survey of Design Methods for Failure Detection in Dynamic Systems. *Automatica,* Vol. 12 (6), pp. 601-611.

Young, P. (1981). Parameter Estimation for Continuous Time Models – A Survey. *Automatica,* Vol. 17 (1), pp 23-29.

APPENDIX 1

JNMM PAPER

Tank Measurement Data Compression for Solution Monitoring

E. C. Miller and J. Howell
Department of Mechanical Engineering, University of Glasgow
Glasgow, United Kingdom

Abstract

A tank measurement data compression scheme is described that has been developed to meet the peculiar needs of nuclear materials safeguards in general, and solution monitoring in particular. The scheme has two stages, a filtering stage followed by a correlate and test stage. The filter might be viewed as an extension of the box car algorithm that enables feature extraction; the correlate substage uses a recursive least squares estimator, and the test stage is based on the Cusum test. The scheme has been developed by testing with data obtained from a commercially operated facility.

Introduction

Solution monitoring has been defined as the "essentially continuous monitoring of solution level, density and temperature in all tanks in the process that contain, or could contain, safeguards-significant quantities of nuclear material."[1,2] As such it is a candidate "defense in-depth" approach, the need for which has been discussed by Andrew.[3] To perform solution monitoring with the data-collection rates thought to be necessary to monitor normal plant activities,[4] it is likely that at least 15,000 points/sensor/month will need to be collected for subsequent analysis. Although not excessively large, it is clear that a large portion of these data would be redundant because a high data-collection rate is only needed when a change in plant activity actually occurs, and this is relatively infrequent. Some form of data compression is therefore desirable from a data-storage viewpoint. In addition, any sensible data compression would also be of benefit to the solution-monitoring system installed if it enables its complexity to be reduced in some way. Solution-monitoring-system development work described to date has not examined this primarily because the data have not been made available to motivate such work. Both Burr and Wangen[1,2] and Scothern and Howell[5] have used similar data to test their respective systems. Either real data, recorded at very infrequent intervals of time, have been used directly, or data have been synthesized; for instance, infrequently recorded real data have been linearly interpolated, sampled at five-minute intervals and corrupted with Gaussian noise. Perhaps it is worth pointing out that these particular infrequently recorded data had already been compressed using a variation of the well-known combined box car and backward slope algorithm,[6] but the justification for using this algorithm is unclear.

The need for data compression is not peculiar to solution monitoring. Data compression is used in many areas and in particular in the process industries.[6–10] However, what might be peculiar about data compression for solution monitoring is that the objective is somewhat different: Process industry data compression is about recording the underlying trends in the data whereas, with its close relationship with nuclear materials accountancy, data compressed for input to a solution-monitoring system should reflect more the quantitative state of the contents of the tank. A good example of the difference arises when a tank is sparged. It might then be necessary to ignore level measurements because level fluctuations would be misleading; level is used to infer volume via a calibration obtained with sparging switched off. Thus any data compression algorithm that is suitable for solution monitoring is likely to have an element of feature extraction in it and is hence unlikely to be generally applicable to the process industries.

This paper describes a data compression scheme (Figure 1) that extracts all the important features of data pertaining to tanks in which liquor is not imported and exported at the same time. The results are output as four separate streams: Stream A con-

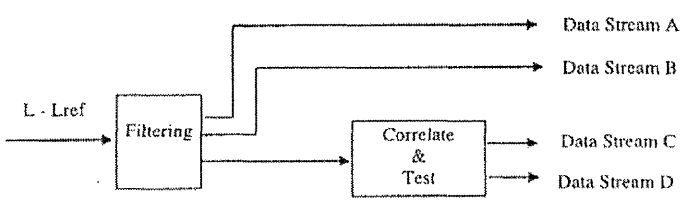


Figure 1. Data reduction scheme

tains times at which sparging occurs, stream B contains spikes that might be viewed as "odd" events, while streams C and D contain only those data records that need to be input into a solution-monitoring system like that developed at the University of Glasgow[1-5] The only difference between streams C and D is in their data format. The scheme would need to be modified to process measurement data obtained from tanks that can have simultaneous inputs and outputs to and from them (e.g. a solvent-extraction-cycle receiving tank), and this is discussed in the conclusions.

The scheme has been developed and tested by analyzing data collected from a number of tanks located in a product storage area of a reprocessing plant. The data were collected using the Ispra VLTM system,[11,12] which is capable of collecting data through six multiplexed channels. Each channel is interrogated over a three-second period, approximately once every 18 seconds; about eight measurements are input and averaged over the three-second period, and the standard deviation can also be calculated. Note that the maximum collection rate of once every 18 seconds is about six times faster than that which is nominally required. Instead of just slowing down the sampling rate, the developers have chosen to provide the facility to compress the data instead using a variation of the box car algorithm.[5,11] Typical data compressions of between 75 percent and 85 percent have been obtained where

$$\text{Data compression} = \frac{\begin{matrix}\text{Total number of}\\ \text{averaged}\\ \text{measurements input}\end{matrix} - \begin{matrix}\text{Total number of}\\ \text{averaged}\\ \text{measurements recorded}\end{matrix}}{\begin{matrix}\text{Total number of averaged}\\ \text{measurements input}\end{matrix}}$$

Note that an average data compression of about 88 percent would give the nominal goal of 15,000 points/sensor/month, whereas the scheme described in this paper has achieved better than 99.9 percent. To avoid confusion, it is worth pointing out that the work described here has largely been carried out using data collected with the developer's facility switched off; that is, the scheme has been developed using data recorded approximately every 20 seconds. Each data record is assumed to contain, where available, level, density and reference dip-tube averaged pressure measurements, plus the standard deviations of each of the level dip-tube averaged pressures. On input, the equivalent level and density are then calculated and added to each record. The data-compression algorithms are then applied to level and density as opposed to dip-tube pressures, because not only are level and density of primary interest, but also data compression based on the level recording eliminates the common influence on all the dip-tube pressures caused by changes in atmospheric pressure.

Typical Measurement Records

A typical tank level history is shown in Figure 2 for a period of time where there are no transfers of material into or out of the tank. Note that the level does not remain constant because the tank is sparged about every seven minutes and there is evaporation. Two sparge cycles that are clearly identifiable in Figure 2 are characterized by a series of about six points: a downward reflection, followed by a sharp upward increase to some point above the true level, followed by a plateau for two to three points that remains above the true level, followed by a short upward trend and then a sharp downward decrease to another reflection before returning around the true mean. Figure 2 also has some temporal deviations or spikes, where the level deviates from and then immediately returns to the true level. Figure 3 shows the data collected while the tank was recirculated and sampled.

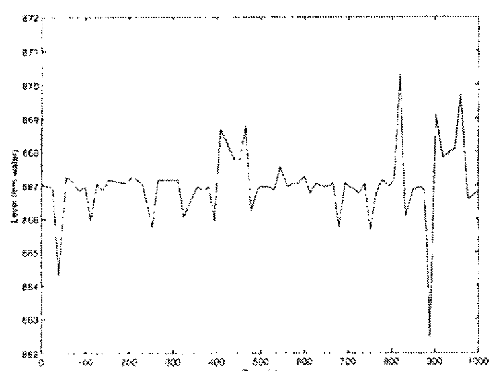


Figure 2. Typical tank level data

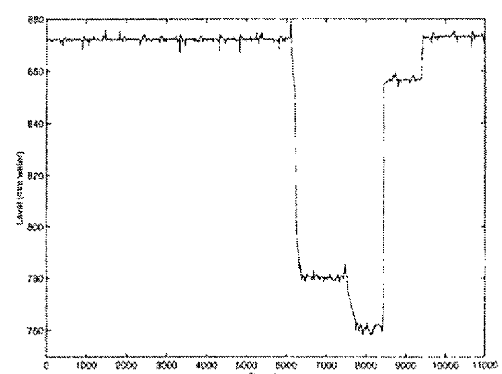


Figure 3. Typical tank level data during recirculation

Several actions can be identified:

1. Recirculate through pump for about 1,800 seconds:
2. Fill and recirculate through the measuring pot;
3. Stop pump and sample measuring pot;
4. Drain measuring pot back to tank.

Application of a Simple Data Compression Algorithm

The application of something like the box car algorithm can add a bias to the data if process noise is one-sided over timescales of interest. As far as solution monitoring is concerned, process noise includes sparging and small temporal deviations, as these are of no interest in the longer term. An example of this one-sided effect is shown in Figures 4 and 5, where the exponentially weighted moving average has been calculated for both tank level data and for tank level data compressed using the box car algorithm with a 3 mm H_2O tolerance, respectively. The faint, highly noisy data are the original level data that fluctuates around 867 mm (approx.); a time constant of $\lambda = 0.1$ was used in the EWMA.[8]

The Scheme

The data compression scheme is composed of two parts (Figure 1), filtering followed by correlation and testing. The aim of filtering is to:

1. Remove unnecessary data points from the main output stream;
2. Output those times at which sparging occurs through stream A;
3. Output those points that pertain to odd events through stream B;
4. Remove those points pertaining to streams A and B from the main output stream.

The main output stream should then contain only data points corrupted with measurement errors as opposed to process noise. Typically, a data compression of 99.3 percent is obtained at this point. A correlate and test stage is then applied so that only the following are output:

1. Those points pertaining to physical activities like recirculation and transfers:
2. Sufficient points to enable an overall, gradual trend to be observed, for instance, a gradual reduction in level as a result of evaporation.

Depending on the activities ongoing on the plant, the final output stream might contain as little as a few tens of points per month.

The Filtering Procedure

The data-filtering algorithm iterates through the time series by making use of two limits: a *proximity* limit, τ_p, and a *change* limit, τ_{ch}. The proximity limit specifies the bounds within which a data point must lie for it to be ignored and the focus of the algorithm moved on to the following point. The change limit seeks to isolate those events that might be of interest to the solution-monitoring system. Thus it specifies the magnitude that the difference between two adjacent points must exceed before it is passed to the output stream. Let the data series to be filtered be defined by $X_1, X_2, X_3, \ldots, X_p$, let the main output stream be $Y_1, Y_2, Y_3, \ldots$, and note that the input series need not be collected at a constant rate. Then the following tests (Figure 6) are applied:

a) No real change: Ignore X_i if $|X_{i+1} - X_i| \le \tau_p$.
b) Important event: X_i and X_{i+1} → Stream Y if
$|X_{i+1} - X_i| > \tau_{ch}$ and $|X_i - X_{i-1}| \le \tau_p$.
c) Sparging: X_{i+1} → Stream A if
$\tau_p < |X_{i+n} - X_i| \le \tau_{ch}$ and $|X_{i+m+1} - X_i| \le \tau_p$,
$\forall_n n : 1 \le n \le m : 3 \le m \le 10$,
d) Important event: X_i and X_{i+m+1} → Stream Y if
$\tau_p < |X_{i+n} - X_i| \le \tau_{ch}$ and $|X_{i+m+1} - X_i| > \tau_{ch}$,
$\forall_n n : 1 \le n \le m : 3 \le m \le 10$;
e) Sudden change in bias: X_i and X_{i+n} → Stream Y if
$\tau_p < |X_{i+n} - X_i| \le \tau_{ch}$, $\forall_n n : 1 \le n \le 11$.

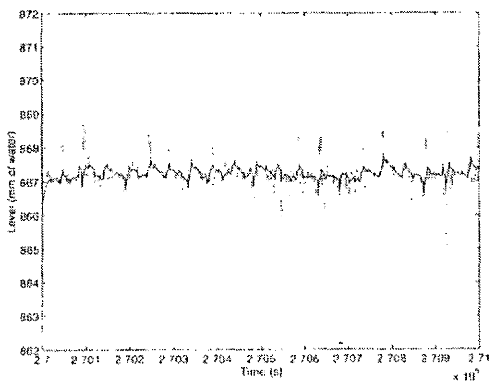


Figure 4. EWMA of original data

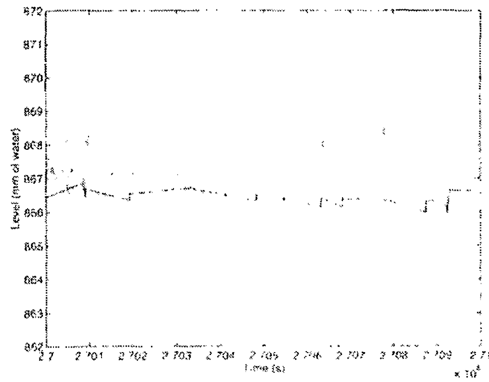


Figure 5. EWMA of compressed data

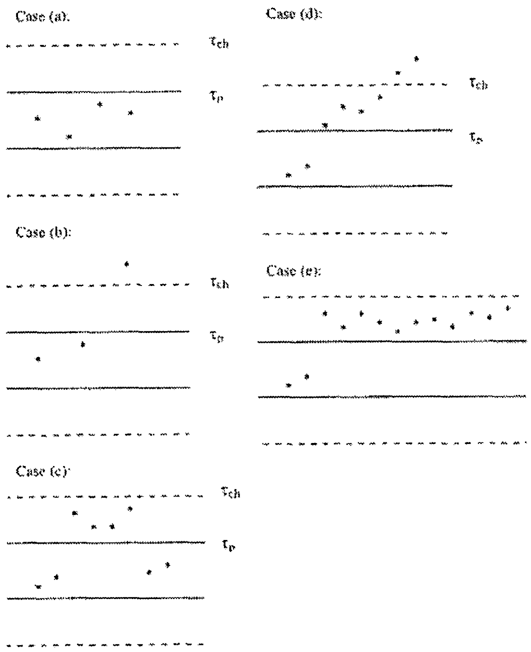


Figure 6. Features identified

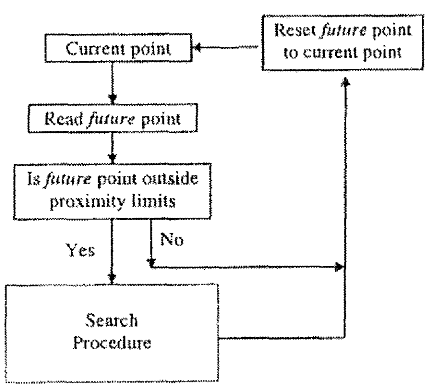


Figure 7. Logical description of filtering algorithm

Odd events are eliminated by looking at the associated level dip-tube standard-deviations σ_1, σ_2, σ_3, ... *after* the other tests have been completed:

a) Odd event: X_k and $X_{k-1} \rightarrow$ Stream B if
$\sigma_k > \tau_\sigma$, where τ_σ is another tolerance.

Note that these tests are not guaranteed to filter out all features that are unimportant. Duplicate records are sometimes output, but these are easily detected and removed.

The main steps of the iterative procedure are shown as flow diagrams in Figures 7 and 8. Starting from the top of the diagram, the algorithm compares the proceeding point with the current point; if the proceeding point is within the proximity limits, it becomes the current point and the next point assumes the mantle of the proceeding point, and so on. If the proceeding point lies outside the proximity limits, the search algorithm is invoked and the current point remains the same until a search is completed. On completion, passing both points to the output stream is important, as it means that both the start point and end point are collected. The proceeding point is checked to ensure that it does not lie within the proximity limits. It is then checked to see if it lies outside the change limit. If it fails to meet either of these criteria, the focus of the algorithm shifts to the next point, i.e., it becomes the proceeding point. This loop continues until either the proceeding point returns to lie within the proximity limits or exceed the change limit. Note that the procedure is more complicated than this to accommodate the maximum loop count of 10 and hence the possibility of a sudden change in bias. Under a certain set of conditions, the level could remain between the proximity and change limits indefinitely, thus causing corruption of the filtered data. If the points lie between the proximity and change limits for 10 points (i.e., for 10 loops of the search loop), it is assumed that the level will never return to within the proximity limit of the current point; the current point and the proceeding point are then passed to the output stream and the proceeding point is then reset as current. The entire process is then started again.

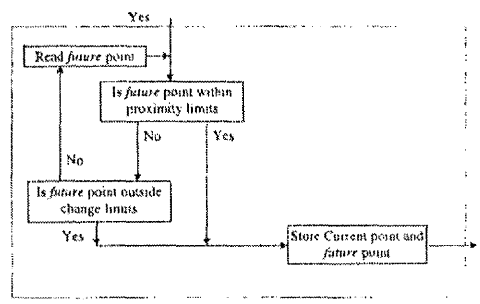


Figure 8. Logical description of search algorithm

Figure 9 shows the resultant filtered output obtained from a small sample of input data. The input data are superimposed as a faint line and each * denotes an output point. Note that a significant number of points are output because of the significant number of temporal deviations that have occurred. To smooth the output further, the filtered data were passed through the filtering algorithm a second time with the proximity limit set to twice that of the first pass and with the same change limit. No additional output streams are required, as all the features of interest should have been passed on the initial filter pass. Figure 10 shows the sample data after being filtered twice: Each point output to the sparging record file is marked as 'o'.

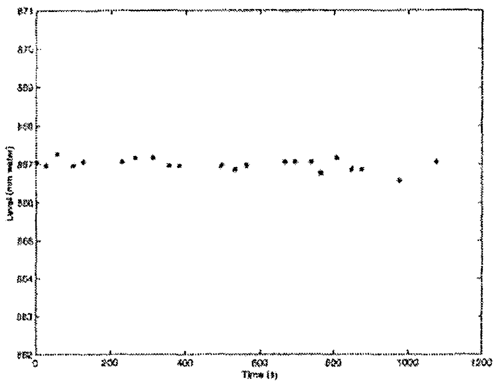


Figure 9. Sample data + output (*) from first filter pass

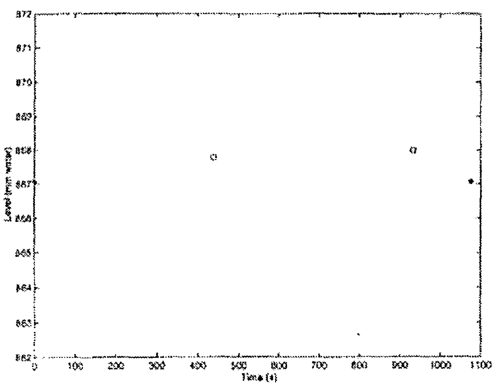


Figure 10. Sample data, output (*) from second filter pass, and sparging (o)

The Correlate and Test Procedure

Although the above achieves a significantly high level of data compression, the level is still likely to be too low for a solution-monitoring system where only relatively infrequent events are of interest. Typically, only 100 points might be of interest in 100,000 points, suggesting a data compression down to, say, one per 1,000, or 99.5 percent. Thus a further procedure is required to extract the desired points.

The correlate and test procedure is based on the observation that, if process noise is ignored, then a level history plot can be represented by a series of straight-line segments, the change points of which correlate with those physical events that might be of interest to a solution-monitoring system. This then leads to the application of an estimator to generate equations for the straight-line segments and the application of a test procedure to detect the change points. The basic scheme is shown in Figure 11. There are several points that need to be taken into consideration in order for this procedure to function in the desired manner:

1. The estimator must be reset upon detection of a change in order for it to construct a new estimate for the process mean;
2. The test procedure has to be switched off whenever the program is restarting (including the initial start) to allow an accurate estimate for the equation of the line. Failure to do so will result in false alarms.

This approach does not output appropriate start and end points for an entire measurement history so, to reduce the effect of process noise, these points are obtained from the EWMA (with $\lambda = 0.1$)[19] that is calculated in parallel with the filter, correlate and test procedures. The start point is then taken to be the 15th point of the EWMA value, and the end point is the last point calculated. This is the only time that calculated data, as opposed to recorded data, are output.

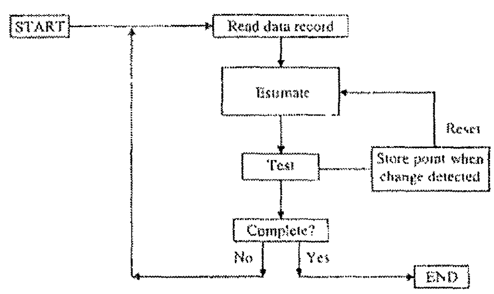


Figure 11. Logical description of the correlate and test procedure

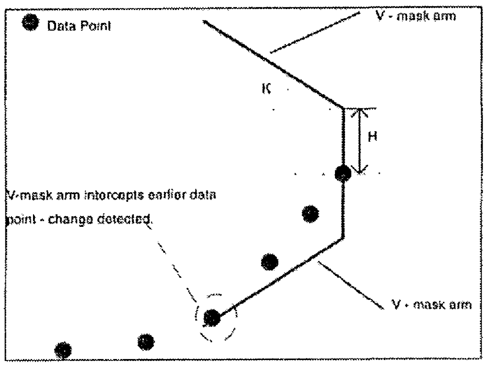


Figure 12. V-mask template

The Correlate Procedure

A recursive least squares algorithm[15] is used to estimate the gradient m and constant c [$m, c : y_i = mz_i + c$; $z_i = t_i - t_1$; t is time] as follows:

$$x_{n+1} = x_n + P_{n+1}\, h_{n+1}\, (y_{n+1} - h^T_{n+1}\, x_n).$$

where:

$y = h^T x + n$

$h = [z\ 1]^T$

$x = [m\ c]^T$

n – noise

$P_{n+1} = P_n - P_n\, h_{n+1}\, (1 + h^T_{n+1}\, P_n\, h_{n+1})^{-1}\, P_n\, h^T_{n+1}$

To apply this, $P_0 = \infty * 2 \times 2$ identity matrix and x_0 has $m = 0$ and c = the first data point. It is necessary to reset the RLS calculation every time a point of change is detected. This is easily achieved by simply resetting P_{n-1} to its initial value and setting X to have $m = 0$ and c = value of process mean at this point.

Due to problems encountered later on when finding m and c at points of change, the values calculated by the above code are stored in a simple ring buffer with a delay between the write and read counters.

The Test Procedure

A modified version of the standard Cusum test[14,16] is used to test for changes. From reference 14, expressions for the upper Cusum (C^+) and the lower Cusum (C^-), when applied to a normalized data point x_i, are as follows:

$$C_i^+ = \max\,[0,\ x_i - (\mu_o + K) + C^+_{i-1}]$$

$$C_i^- = \max\,[0,\ (\mu_o - K) - x_i + C^-_{i-1}]$$

where μ_o is the target process mean and K is a constant.

If either C_i^+ or C_i^- exceeds appropriate tolerances, then the process mean is deemed to be changing. This procedure is often represented by a V-mask (Figure 12) that is applied at every point of the data; if one of the arms of the V-mask intercepts a data point then a change in the mean is deemed to have taken place. The values of H and K relate to the vertical height and the angle of the arms of the V-mask respectively. If $K = 0$, the arms are at right angles. Here, the Cusum test is applied to the data without normalization, and hence the value of H is related to the standard deviation, σ, of the input data:

$$H = h * \sigma$$

where h = a constant ($1 < h < 5$). Note that standard deviation estimates are output by the VLTM system directly.

This procedure has to be modified to account for the fact that the test is not on a process mean that is assumed to be constant, but on whether or not the data have deviated from a straight-line trajectory. Thus μ_o is replaced by $mz_i + c$.

Due to the nature of the V-mask test, the time at which the change is detected is at least one time step after the start of the actual physical change. There are two ways of procuring a more accurate estimate, the choice of which is dependent on the output that is desired by the user.

1. Store both the gradient and the constant of the line prior to the point of change and, once the estimate of the subsequent line has been completed, calculate the intersection of both lines to obtain a new point of change that might not exist within the original data;
2. Temporarily store the data points so that an appropriate point can be recalled and used as the change point. Thus the point of change will exist within the input data set.

In this instance the latter option was chosen because the output of genuine data points was preferred.

The V-mask requires careful calibration for it to process data successfully. Many factors affect its operation. For example, how soon after least squares should it be restarted? Answers to these and other questions are dependent on the density of the data points to be processed. The VLTM data sets are of relatively low density, with substantial changes in the process mean occurring within five or more points *before* filtering.

Figure 13 shows the points output over a one-month period (approx.) superimposed over the filtered data because the input data set is so large. Figure 14 is a close-up of the recirculation period. Note that the end points can be used to estimate the effect of evaporation and that there are sufficient points to extract all the important features of the recirculation/sampling.

Conclusions

A tank measurement data compression scheme has been described that has been developed to meet the special needs of nuclear materials safeguards in general and solution monitoring in particular. Depending on the amount of activity that is associated with the tank, a high degree of data compression can be obtained. The outputs from a number of recorded data sets have been input into the solution-monitoring system developed by Scothern and Howell[1] and, in every instance, the recirculation/sampling activities were detected and diagnosed.

The scheme has been developed to compress data pertaining to tanks whose imports/exports are in the form of batches. It is a moot point whether it would work without modification on, for instance, the receiving tank of a solvent-extraction cycle, because here the level history is likely to have some form of sawtooth profile. In such cases, the V-mask test might produce erroneous outputs if the vertices of the profile are too sharp. There are a number of ways of accommodating this, but modification is not thought to be worthwhile until appropriate real test data are available to work with.

Acknowledgments

This work was funded by the U.K. Department of Trade and Industry through the Safeguards R&D Programme in support of International Atomic Energy Agency Safeguards. The results of this work may be used in the formulation of government policy, but views expressed in this report do not necessarily represent government policy.

References

1. Burr, T.L., and L. Wangen. "Development and Evaluation of Methods for Safeguards Use of Solution Monitoring Data," LA-13185-MS. Los Alamos National Laboratory, 1996.
2. Burr, T.L., and L. Wangen. "Enhanced Safeguards via Solution Monitoring," LA-13186-MS. Los Alamos National Laboratory, 1996.
3. Andrew, G. "Safeguards: Changes and Challenges" *Journal of Nuclear Materials Management* 26(2), 1998, pp. 12–16.
4. Franssen, F. "Tank Data Acquisition and Evaluation in a Large-scale Reprocessing Plant." 17th ESARDA Annual Symposium on Safeguards and Nuclear Material Management, Aachen, Germany, 1995.
5. Scothern, S.J., and J. Howell. "A

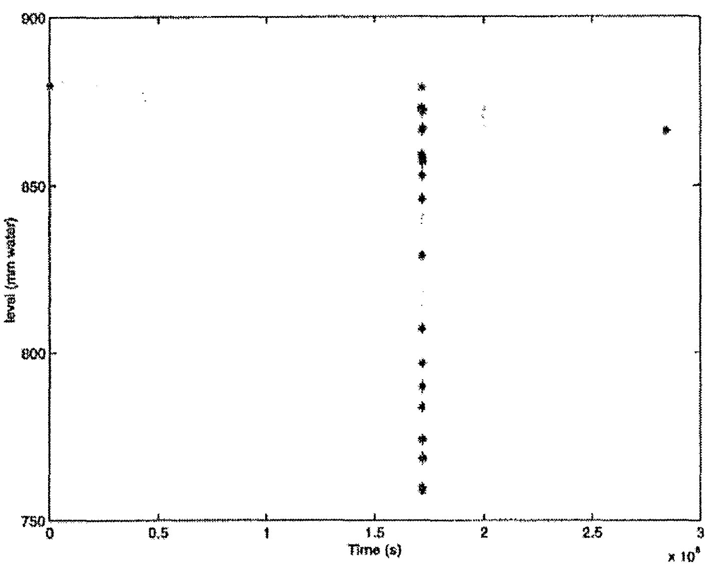


Figure 13. V-mask output superimposed on filtered data

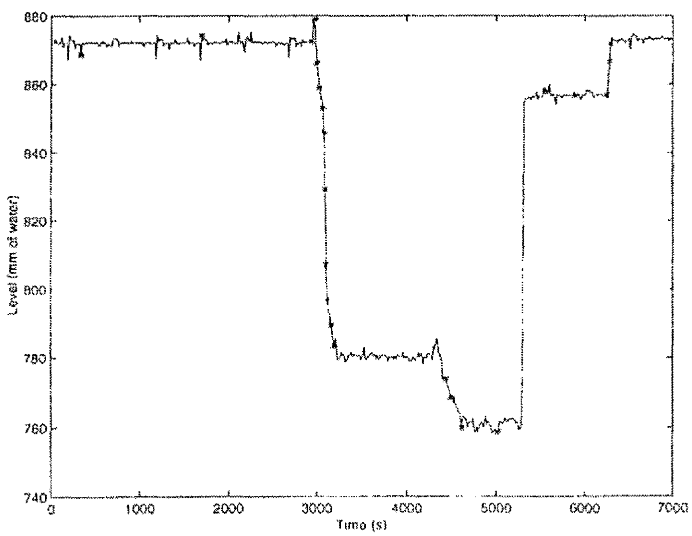


Figure 14. V-mask output superimposed on recirculation

Physical-model-based Diagnostic Aid for Safeguarding Nuclear Material in a Liquor Storage Facility," *Journal of Nuclear Materials Management* 25(4), 1997, pp. 20–29.
6. Hale, J.C., and H.L. Sellars. "Historical Data Recording for Process Computers." *Chemical Engineering Progress*, Nov. 1981, pp. 38–43.
7. Bristol, E.H. "Swinging Door Trending: Adaptive Trend Recording?" *Advances in Instrumentation* 45(2), 1990, pp. 749–754.
8. Cheung, J.T.Y., and G. Stephanopoulos. "On the Detection and Representation of Trends." *Advances in Instrumentation* 45(2), 1990, pp. 755–774.
9. Cheung, J.T.Y., and G. Stephanopoulos. "Representation of Process Trends, Parts I & II." *Computers & Chemical Engineering* 14(4–5), April–May 1990, pp. 495–539.
10. Watson, M.J., A. Liakopoulos, D. Brzakovic, and C. Georgakis. "Practical Assessment of Process Data Compression Techniques." *Industrial & Engineering Chemistry Research* 37(1), Jan. 1998, pp. 267–274.
11. Landat, D., M. Caviglia, and B.A. Hunt. "Characterization and Use of the Volume Long-Term Monitoring Device, VLTM," Technical Note N. I.98.92, JRC Ispra, May 1998.
12. Landat, D., B.A. Hunt, W. Koehne, F. Franssen, and D. Hope. "Portable Measurement Equipment for Unattended Process Monitoring and Verification of Liquids in Tanks," *Proceedings of the 38th Annual Meeting of the Institute of Nuclear Materials Management*, Phoenix, Ariz., 1997.
13. Landat, D., B.A. Hunt, G. Baratti, and M. Galli. "Development and Application of Portable Inspection Monitoring Systems for Tank Calibrations and Unattended Verification of Volume Measurements." *Proceedings of the 19th ESARDA Annual Symposium on Safeguards and Nuclear Material Management*, Montpellier, France, 1997.
14. Montgomery, D.C. *Introduction to Statistical Quality Control, 3rd Edition*. Chichester: Wiley, 1996.
15. Graup, V. *Identification of Systems*. Van Nostrum, 1972.
16. Chatfield, C. *Statistics for Technology: A Course in Applied Statistics, Third edition (Revised)*. London: Chapman and Hall, 1996.

John Howell graduated with a degree in control systems and subsequently joined the Control and Dynamics Group of the U.K. Atomic Energy Authority. He worked on various aspects of the fast reactor project, including the prototype fast reactor power station, the modelling and control of an associated solvent-extraction plant, and near-real-time materials accountancy. He joined the lecturing staff of the University of Glasgow in 1984 and received a doctorate in model-based fault detection in 1990. His main area of research is in model-based fault detection, diagnosis and anomaly resolution.

Euan Miller graduated with a degree in mechanical engineering from University of Glasgow in 1998. He is now a research assistant at University of Glasgow researching into solution monitoring.

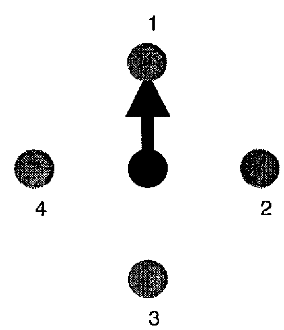
APPENDIX 2

DATA ACQUISITION PROBLEMS

This appendix describes the problems that may occur when a hardware-based data compression system is included in the design of the tank level measurement system.

A2.1 Time Issues

Consider the 4-input scanivalve, which is connected to four tanks to measure the level dip-tube pressures in Pa. The valve is 'illustrated' below:

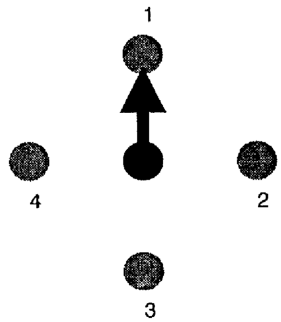


The 'pointer' rotates with a total time to move between the ports on the valve of ten seconds. The pressure signal is measured over 5 seconds, the rest being used for rotation and settling times. Thus a new averaged reading for each tank is obtained every ten seconds.

The valve rotates in a clockwise direction (reading ports 1,2,3,4). The collected data is held in buffer, with each new measurement over-writing the previous measurement for that tank. Every thirty seconds the contents of the buffer are recorded in a data file.

What happens as the reading head rotates is illustrated on the following pages.

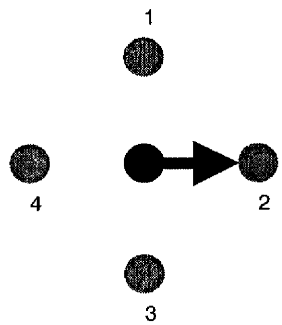
t = 0



Reading: $P_1(0)$

Buffer: $P_1(0)$, $P_2(?)$, $P_3(?)$, $P_4(?)$

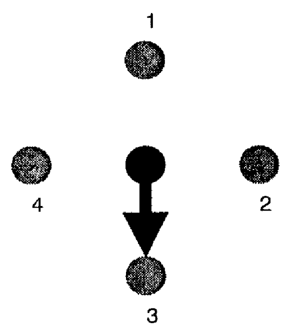
t = 10



Reading: $P_2(10)$

Buffer: $P_1(0)$, $P_2(10)$, $P_3(?)$, $P_4(?)$

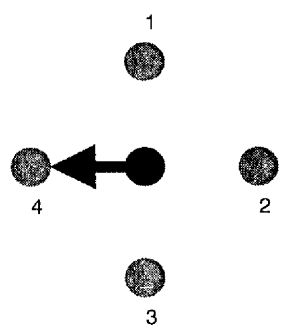
t=20



Reading: $P_3(20)$

Buffer: $P_1(0)$, $P_2(10)$, $P_3(20)$, $P_4(?)$

t=30

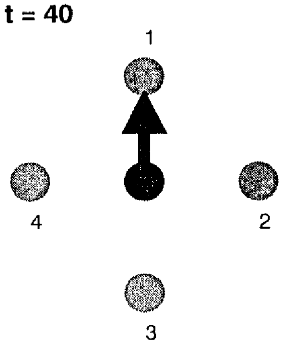


Reading: $P_4(30)$

Buffer: $P_1(0)$, $P_2(10)$, $P_3(20)$, $P_4(30)$

Written to file: $P_1(0)$, $P_2(10)$, $P_3(20)$, $P_4(30)$

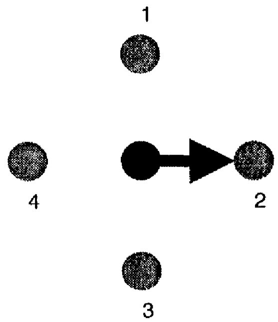
t = 40



Reading: $P_1(40)$

Buffer: $P_1(40)$, $P_2(10)$, $P_3(20)$, $P_4(30)$

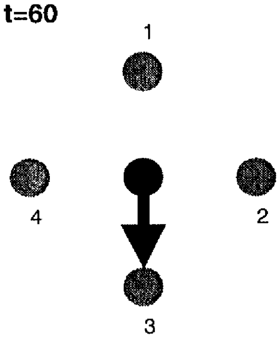
t = 50



Reading: $P_2(50)$

Buffer: $P_1(40)$, $P_2(50)$, $P_3(20)$, $P_4(30)$

t=60



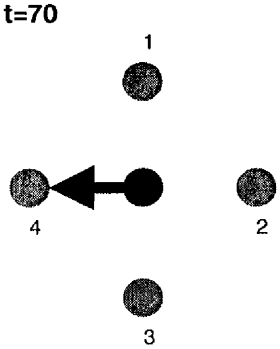
Reading: $P_3(60)$

Buffer: $P_1(40)$, $P_2(50)$, $P_3(60)$, $P_4(30)$

Written to file: $P_1(40)$, $P_2(50)$, $P_3(60)$, $P_4(\mathbf{30})$

P_4 is unchanged but still written to file.

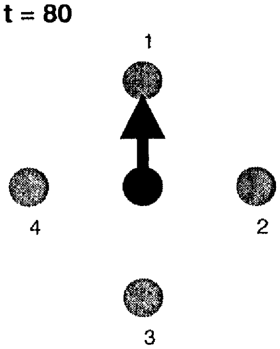
t=70



Reading: $P_4(70)$

Buffer: $P_1(40)$, $P_2(50)$, $P_3(60)$, $P_4(70)$

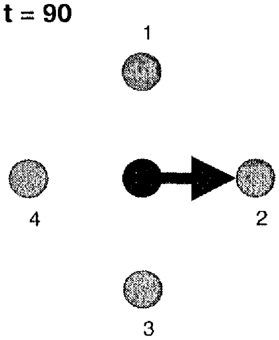
t = 80



Reading: $P_1(80)$

Buffer: $P_1(80)$, $P_2(50)$, $P_3(60)$, $P_4(70)$

t = 90



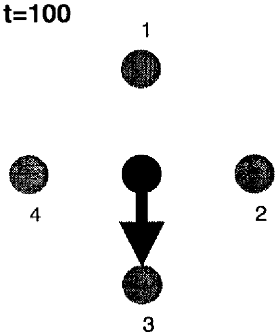
Reading: $P_2(90)$

Buffer: $P_1(80)$, $P_2(90)$, $P_3(60)$, $P_4(70)$

Written to file: $P_1(80)$, $P_2(90)$, $P_3(\mathbf{60})$, $P_4(70)$

P_3 is unchanged but still written to file.

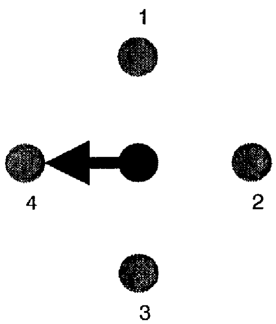
t=100



Reading: $P_3(100)$

Buffer: $P_1(80)$, $P_2(90)$, $P_3(100)$, $P_4(70)$

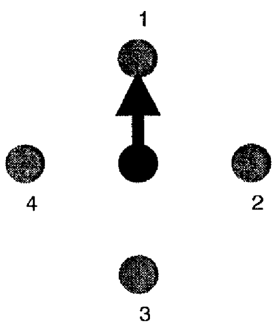
t=110



Reading: $P_4(110)$

Buffer: $P_1(80)$, $P_2(90)$, $P_3(100)$, $P_4(110)$

t = 120



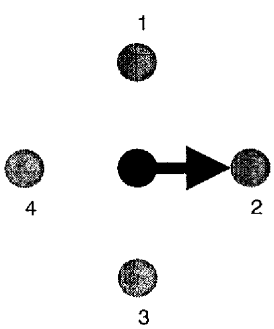
Reading: $P_1(120)$

Buffer: $P_1(120)$, $P_2(90)$, $P_3(100)$, $P_4(110)$

Written to file: $P_1(120)$, P_2(**90**), $P_3(100)$, $P_4(110)$

P_2 is unchanged but still written to file.

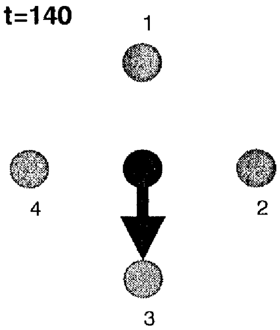
t = 130



Reading: $P_2(130)$

Buffer: $P_1(120)$, $P_2(130)$, $P_3(100)$, $P_4(110)$

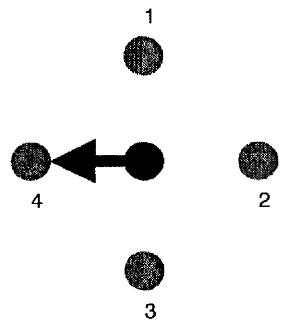
t=140



Reading: $P_3(140)$

Buffer: $P_1(120)$, $P_2(130)$, $P_3(140)$, $P_4(110)$

t=150



Reading: $P_4(150)$

Buffer: $P_1(120)$, $P_2(130)$, $P_3(140)$, $P_4(150)$

Written to file: P_1**(120)**, $P_2(130)$, $P_3(140)$, $P_4(150)$

P_1 is unchanged but still written to file.

Note the following effects:

- frozen points: the same record duplicated 30 seconds later;
- misalignment of data points in time. The measurements of two tanks are ten and twenty seconds old when the data is recorded i.e. for certain tanks the time recorded will be different from the time collected.

This has the following implications:

- plateaux in the middle of transfers due to frozen points;
- inaccurate starts to transfers if coincides with frozen point;
- transfers extended due to sliding end points of transfers.

A2.2 Quantisation

Frozen points can be eliminated easily if it is clear that the chances of two records, recorded one after the other, being identical is extremely remote. Unfortunately the averaged pressures are only recorded to an accuracy that reflects the accuracy of the measurement and the 'noise' is lost.

For example the pressure readings for tank 1 may be:

Time	Pressure (Pa) Calculated	Pressure (Pa) Recorded
0	14910.12	14910
30	14910.12	14910
60	14911.32	14911
90	14910.21	14910
120	14910.34	14910

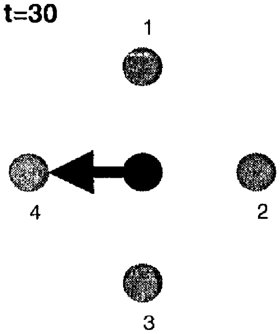
Table A2.2.1 Example pressure readings showing quantisation

Is the second point a frozen point or a new calculation? Ditto for the fifth point. Without recording to at least 2 significant figures (for statistical reasons) it is impossible to tell. Thus the second point is a frozen point, the fifth being a new calculation. Ideally the data should be recorded to at least 2 figures more than accuracy.

A2.3 Speed Issues

In order to understand the discussion that follows consider the following example : transfer tank 4 to tank 3 with a pipe delay is of low magnitude (20 secs).

t=30

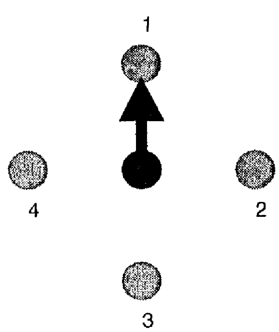


Reading: $P_4(30)$

Buffer: $P_1(0)$, $P_2(10)$, $P_3(20)$, $P_4(30)$

Written to file: $P_1(0)$, $P_2(10)$, $P_3(20)$, $P_4(30)$

t = 40

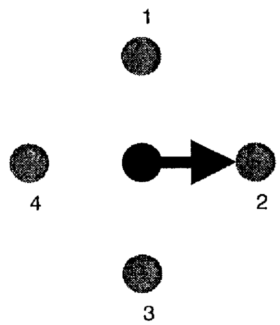


Reading: $P_1(40)$

Buffer: $P_1(40)$, $P_2(10)$, $P_3(20)$, $P_4(30)$

Transfer starts.

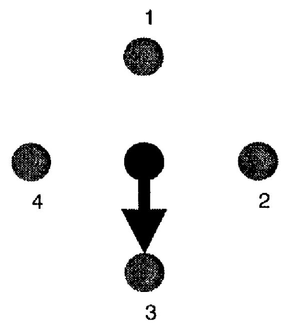
t = 50



Reading: $P_2(50)$ Transfer Continues

Buffer: $P_1(40)$, $P_2(50)$, $P_3(20)$, $P_4(30)$

t=60



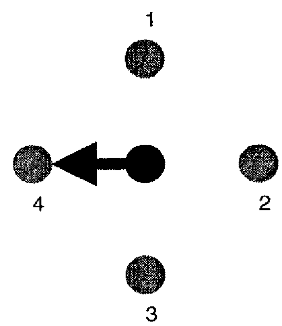
Reading: $P_3(60)$ Transfer Continues

Buffer: $P_1(40)$, $P_2(50)$, $P_3(60)$, $P_4(30)$

Written to file: $P_1(40)$, $P_2(50)$, $P_3(60)$, P_4(**30**)

P_4, no indication of transfer written to file but Tank 3 shows receiving

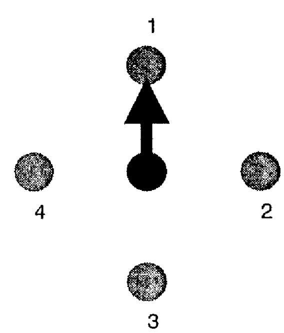
t=70



Reading: $P_4(70)$ Transfer Continues

Buffer: $P_1(40)$, $P_2(50)$, $P_3(60)$, $P_4(70)$

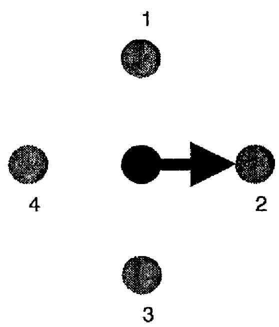
t = 80



Reading: $P_1(80)$ Transfer Continues

Buffer: $P_1(80)$, $P_2(50)$, $P_3(60)$, $P_4(70)$

t = 90



Reading: $P_2(90)$ Transfer Continues

Buffer: $P_1(80)$, $P_2(90)$, $P_3(60)$, $P_4(70)$

Written to file: $P_1(80)$, $P_2(90)$, P_3**(60)**, $P_4(70)$

First indication that tank 4 is transferring written to file.

Admittedly this example is convoluted as pipe delays are usually of the magnitude of minutes rather than seconds. Thus with tanks connected to the same scanivalve the following are possibilities (and may both occur on the same transfer):

1. target starts to receive before source is indicated to be transferring.
2. target finishes receiving before the source has finished transferring.

If the source and target are connected to different scani-valves then more problems may be encountered. If the source is on the faster valve (less tanks or faster rotation) then the problems should be minimised. However if the target is on the faster valve, the problems indicated above can be exacerbated. Figure A.1.3.1 is a plot of actual data of a transfer from the JA6 system. The target tank is on a faster valve than the source tank. Note that the target tank trace has been shifted to allow easy comparison.

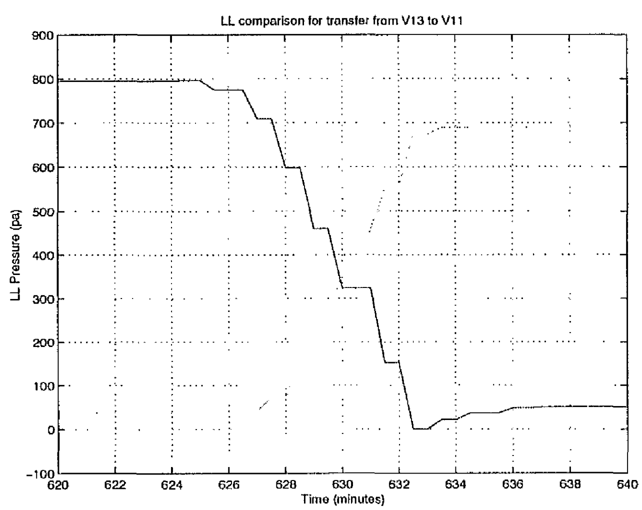


Figure A2.3.1 Comparison of transfer from source to target tank

In this instance all is well as the source tank starts and stops before the target does. However, if the transfer was over a longer period of time, the target tank may overtake the source tank (i.e. the target tank has received more material than the source tank has transferred) and finish before the source tank has completed transferring due to the speed differences between the valves.

APPENDIX 3

This appendix describes an assessment of the JA6 monitoring system currently installed in the product storage area of the Tokai-Mura Reprocessing Plant.

JA6 ISSUES

Ideally, the performance of the JA6 monitoring system should be assessed by inputting known analogue signals into it. Unfortunately such data was not available for the JA6 system so an analogue signal was effectively re-constructed from data collected using the Ispra VLTM system. The effect that the peculiarities in Appendix 1 have on recorded dip-tube pressure measurements was then examined by processing this data as if it had been collected via the scanivalves instead.

VLTM collects for 3 seconds every 18 seconds so a data point at say 18 secs really pertains to 16.5±1.5 secs whereas JA6 collects for 5 seconds every 40 seconds. Hence simulated JA6 data was obtained by:

- Noting that all VLTM time records should have 1.5 deducted from them;
- Interpolating the VLTM data to generate values at 40 second intervals (starting at 37.5 seconds);
- Recording these values at 30 second intervals assuming a 'zero order hold facility'.

Some typical VLTM data 'drawn' as an analogue signal is shown in Figure A3.1. Figure A3.2 shows the (scanivalve) output that would be produced if this 'analogue signal' was to be input into it, Figure A3.3 shows the JA6 data that would be collected and Figure A3.4 compares this data with the original signal. The most important feature in this comparison relates to the lack of definition during the three instances when the tank was sparged. This definition is needed to justify the exclusion of data points during sparging.

An example of the time shift imposed by the scanivalve is shown in Figure A3.5. Visually, during plant activities like a recirculation (Figure A3.6), JA6 appears to track the data adequately. However note the plateau on the down leg (enlarged in Figure A3.7), which was caused by the insertion of a 'frozen' point. Although visually insignificant such instances cause difficulties when applying automated methods. Similarly note the delay on the final up leg (enlarged in Figure A3.8).

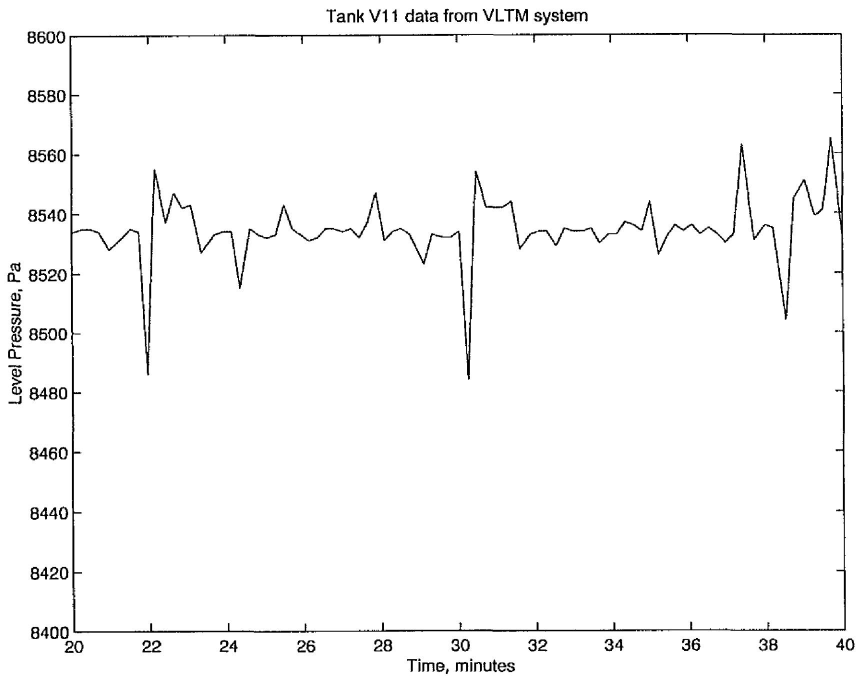


Figure A.3.1 Sample of vltm data from tank 11.

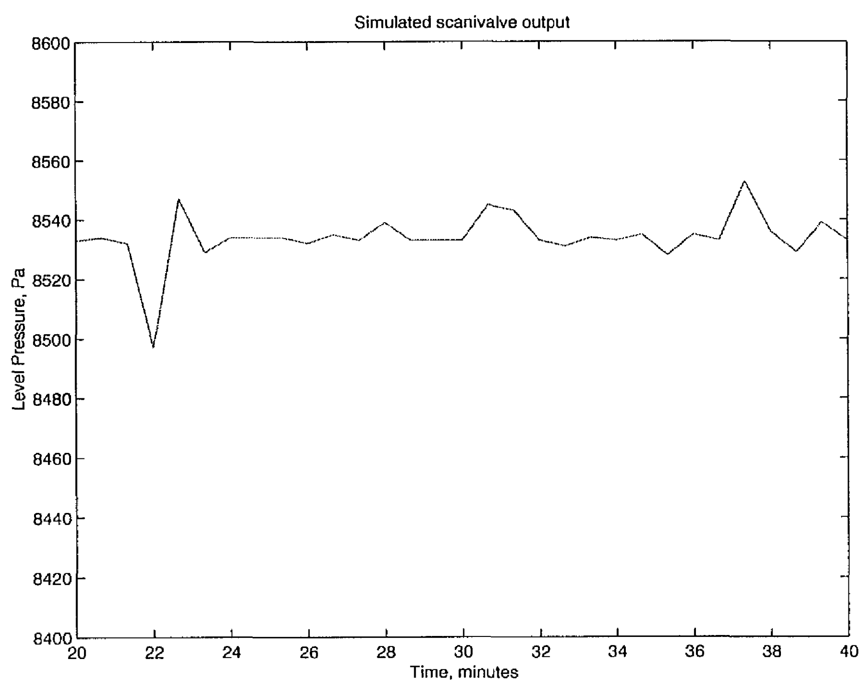


Figure A.3.2 Simulated scanivalve data

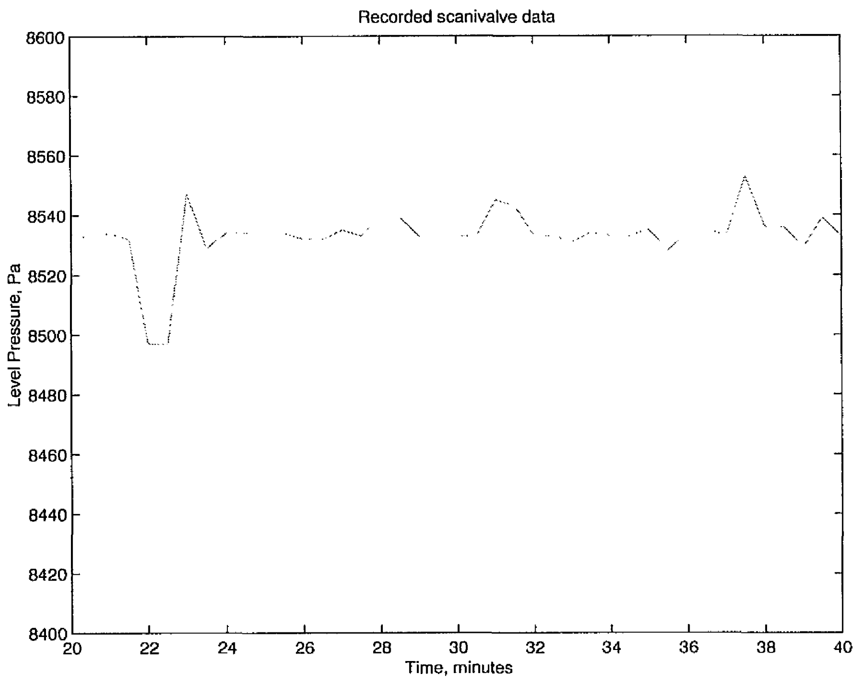


Figure A.3.3 Recorded data

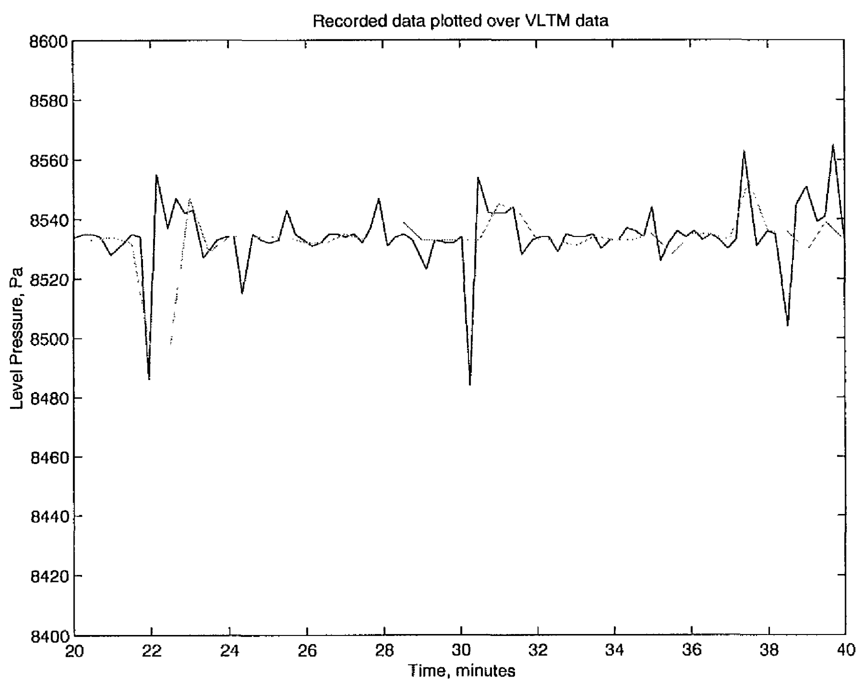


Figure A3.4 Comparison of recorded data with VLTM data

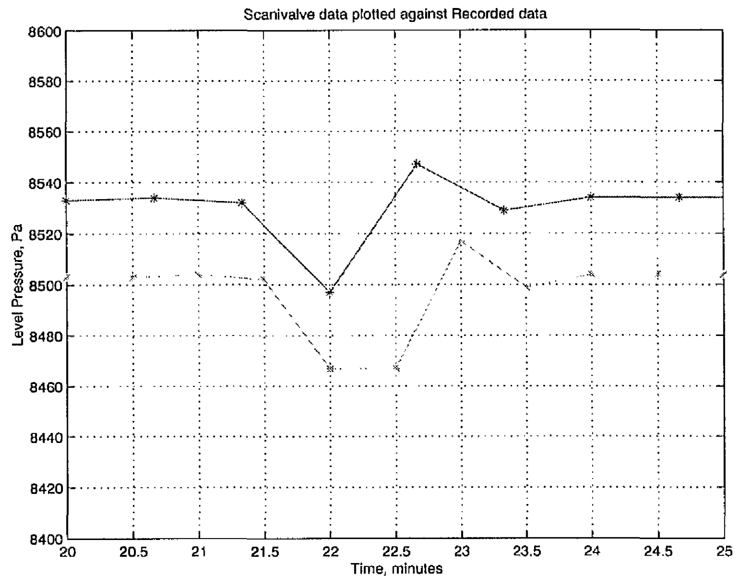


Figure A3.5 Recorded data compared with scanivalve data showing time shift.

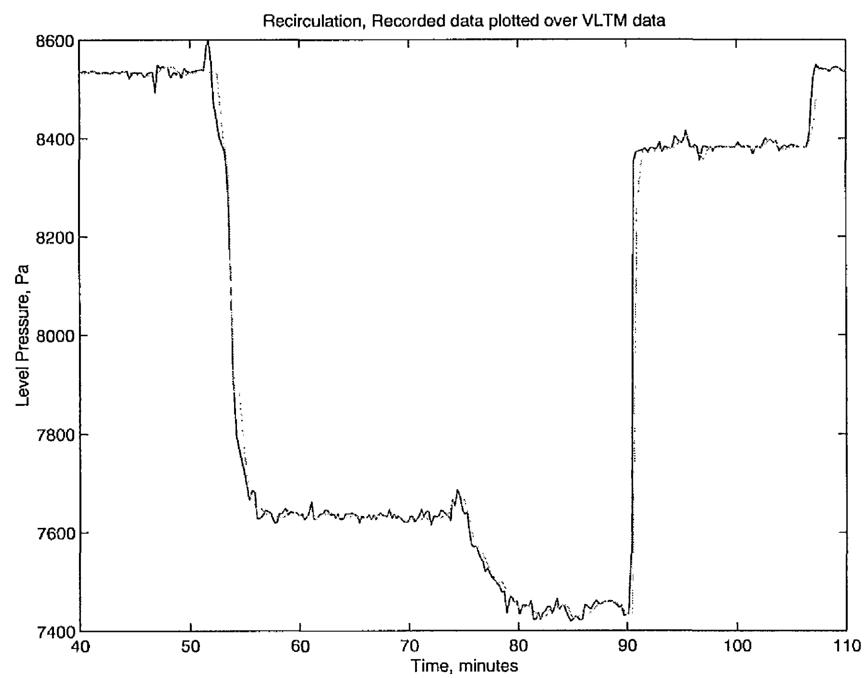


Figure A3.6 Recirculation in tank V11, VLTM and recorded data.

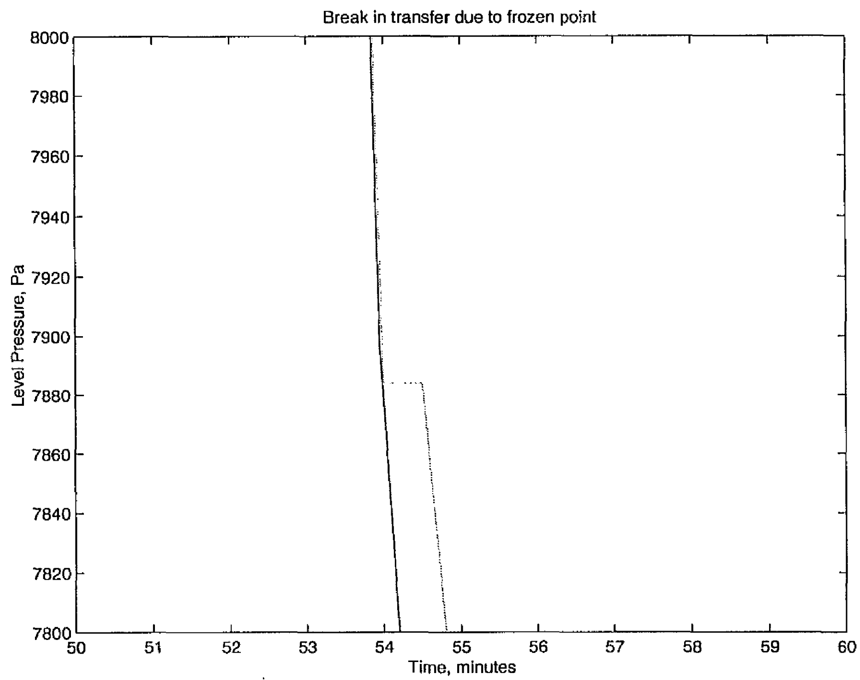


Figure A3.7 Break in continuous transfer due to frozen point

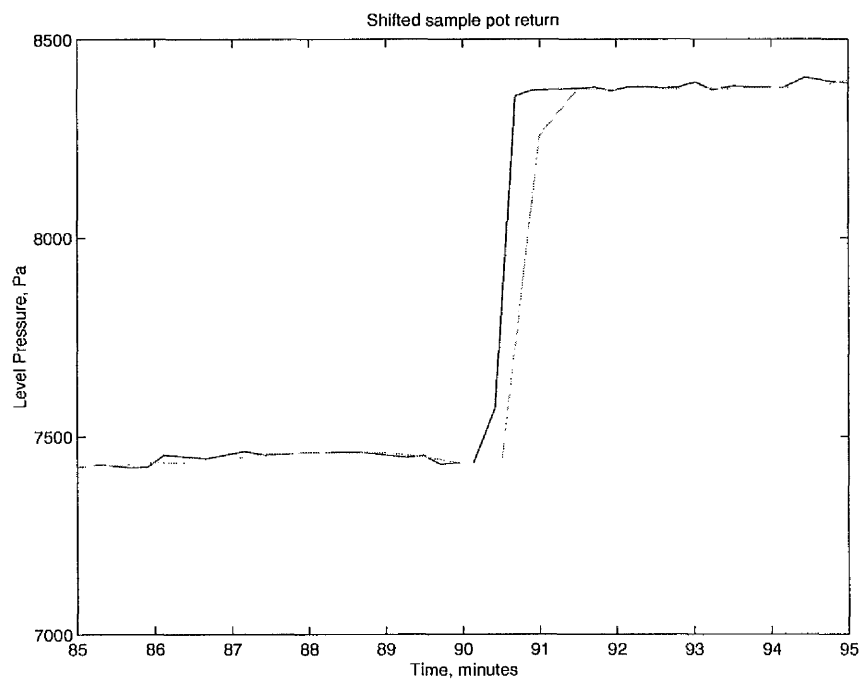


Figure A3.8 Shifted sample-pot return

APPENDIX 4

This appendix describes the affect on data collection software performance of varying a single parameter.

VLTM PARAMETER COMPARISON

The VLTM system has a number of user specified parameters which relate to data compression. Figures A4.1 and A4.2 are plots of actual VLTM level measurement data that are presented here to show the effect of varying a single key parameter 'Alarm Threshold'. The difference can be clearly seen.

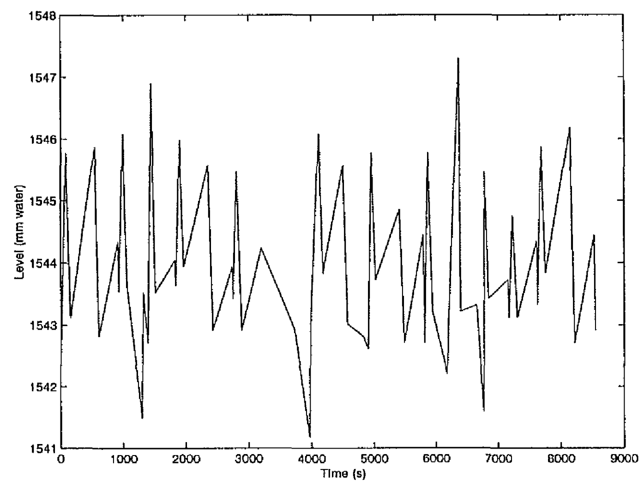


Figure A4.1 Level measurements with alarm threshold of 0.03 kPa

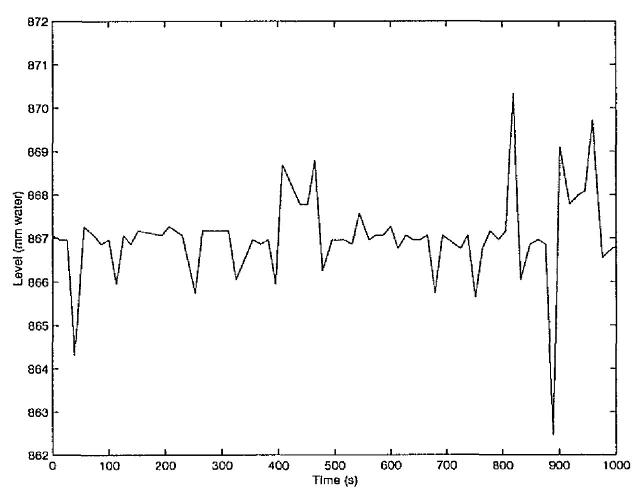


Figure A4.2 Level measurements with alarm threshold 0.0 kPa

ADDENDUM

Department of Trade and Industry

SRDP-R279
UK D1145

EVALUATION OF PROCESS INFORMATION TO OBTAIN ADDITIONAL SAFEGUARDS ASSURANCES IN REPROCESSING PLANTS

J Howell
E C Miller

June 2001

UK Safeguards Support for the IAEA

The UK Safeguards Support Programme for the IAEA is managed by the UKAEA on behalf of the UK Department of Trade and Industry. Further information may be obtained from the Manager, Safeguards Programme Letter, Nuclear Materials Control Office, UKAEA, Harwell, Oxfordshire OX11 0RA, United Kindom

SRDP-R279
UK D1145

EVALUATION OF PROCESS INFORMATION TO OBTAIN ADDITIONAL SAFEGUARDS ASSURANCES IN REPROCESSING PLANTS

J Howell
E C Miller

March 2001

This work was funded by the UK Department of Trade & Industry through the Safeguards R&D Programme in Support of IAEA Safeguards.

Department of Mechanical Engineering
University of Glasgow
Glasgow
G12 8QQ

Telephone: +44 141 330 4070

EVALUATION OF PROCESS INFORMATION TO OBTAIN ADDITIONAL SAFEGUARDS ASSURANCES IN REPROCESSING PLANTS

J Howell
E C Miller

EXECUTIVE SUMMARY

Due to the large throughput in some reprocessing plants, safeguards measures in addition to those that are currently utilised are needed to provide "additional assurances" that the plant is operating as declared. This report outlines the concept of one such measure: the evaluation of process information to confirm that a facility is operating in a way declared by the operator. When doubts do arise, there must be sufficient evidence to justify the taking of this position. In addition, any approach must address the fact that inspector resources are finite, there is a requirement that their effort should be kept to a minimum. The amount of data analysis needed to confirm that operation is as declared will be significant if a high level of automation is to be achieved. Thus there should be considerable emphasis on the representation, storage and presentation of the data and of the evaluations. The concept outlined here seeks to meet these objectives. The proposed approach can be viewed as a candidate element of a "defence in-depth" approach that seeks to compliment accountancy-centred activities. This report focuses on the process area, brief descriptions of possible approaches for accountancy tanks, storage areas and finishing plants are given in an appendix.

An initial assessment would be based largely on the operator's declaration, on data derived from a combination of pressure and temperature measurement systems installed in vessels, and on accountancy tank data. Further investigation might involve data from neutron detectors, XRF detectors, samples, flow meters, flow rate sensors, a measurement of the heat input into the concentrator and so on.

The approach described here is based on the concept known as *analytical redundancy*. Put simply, a plant can be viewed as a number of units that are connected together: if an incident occurs in one unit, its effect is likely to be observed in others so that any local effects must correlate with effects observed elsewhere. Detection and diagnosis is performed by looking at the entire picture, which makes falsification difficult. Put more formally, analytical redundancy exploits the inherent static and dynamic relationships among the measured variables. In other words one makes use of a *mathematical model*. Mathematical modelling is not new to nuclear materials safeguards; for instance materials accountancy is based on a discrete model of the hold-up in a plant. The concept here seeks to extend *this* model, and not to propose a quantum leap in the type of model adopted. Thus the concept is based on the application of a continuous materials account (or *balance*) to the plant; this account can be decomposed into a set of materials balances, one balance per plant unit, and each individual balance would be based on the transfers into and out of that unit. Some of these transfers would also appear in the materials balances pertaining to the plant units that are connected

directly to this unit, because transfers are made from one unit to another. The balance equations would be connected. By analysing these transfers, the underlying goal is to confirm that the nuclear material contained in a plant can be distributed in such a way that it correlates with an inspector's perception of what is going on in a plant, and with the measurements that are available. This goal would be reassessed continually because reprocessing plants operate in time. To do this, use is made of computer simulations to continually predict the distribution of material by solving these balances; this is akin to a continuous version of a discrete simulation of the hold-up in a plant i.e. of a running book inventory. Any disagreements that arise would then be output as *events* in what is known as an *event list*. Clearly in the process of performing these actions other goals arise, which increase the system's capability.

The two basic elements that are required to achieve these goals are plant data and evaluation tools. In addition the storage and manipulation of data and knowledge are central if a high level of automation is to be achieved. Thus it is envisaged that any implementation would contain a kernel composed of two databases, a real-time database and an object oriented database, both of which are available commercially. Appropriate, time-stamped data would be stored in the real-time database whilst other knowledge would be stored as objects. Co-ordinated by a real-time software procedure known as the *Executive*, evaluation tools would estimate the transfers, perform the computer simulation, compare what is predicted with what is observed, and corroborate any conclusions reached by referring to other data (knowledge) sources.

The methodology is largely unproven, based on software developments that are evolving all the time, and would require an extensive programme of work to ensure its success on a commercial facility. Thus the conclusions contain recommendations as to how the approach should be implemented. Considerable thought has gone into ensuring that, having invested so much in its implementation, the evaluation tools can be evolved so that the system matches up to expectations. It is important that appropriate data is collected; although the analytical tools can evolve, the data cannot i.e. one cannot go back and collect the same data again. Similarly it is important that both the structure of the databases and the specification of their numerous data fields are carefully thought about; it is difficult to re-build a database once it has started to be used.

ACKNOWLEDGEMENT

The authors would like to thank Dr R. Abedin-Zadeh, Dr M. Ehinger and Ms T. Renis for useful discussions during the preparation of the report.

CONTENTS

APPENDICES

FIGURES

TABLES

EVALUATION OF PROCESS INFORMATION TO OBTAIN ADDITIONAL SAFEGUARDS ASSURANCES IN REPROCESSING PLANTS

J Howell
E C Miller
Department of Mechanical Engineering
University of Glasgow, Glasgow, UK

1. INTRODUCTION

Due to the large throughput in some reprocessing plants, safeguards measures in addition to those that are currently utilised are needed to provide "additional assurances" that the plant is operating as declared. This report outlines the concept of one such measure: the evaluation of process information to confirm that a facility is operating in a way declared by the operator. When doubts do arise, there must be sufficient evidence to justify the taking of this position. Although there might be many ways in which some level of assurance can be gained, few would have the capacity to generate appropriate evidence when needed. In addition, any approach must address the fact that inspector resources are finite, there is a requirement that their effort should be kept to a minimum. The amount of data analysis needed to confirm that operation is as declared will be significant if a high level of automation is to be achieved. Thus there should be considerable emphasis on the representation, storage and presentation of the data and of the evaluations. The concept outlined here seeks to meet these objectives. Although various aspects might have been examined in the past [1-3], it is largely unproven, based on software developments that are evolving all the time, and would require an extensive programme of work to ensure its success on a commercial facility.

The report contains an outline, a description of how the evaluation system would function, some details about its possible implementation, some comments that largely derive from questions that have been posed by others, and conclusions, in which a route to implementation is recommended. The report is intended to provide a broad overview of the approach, whilst the details and some examples are given in the appendices. The evaluation system would be composed of a large number of tools. Each tool is classified as being a member of one of eight toolboxes. Each toolbox is outlined in the main text and described in detail in Appendix 5. Results from various case studies are given in Appendix 6 to show how the approach would work when faced with various incidents.

2. OUTLINE

2.1 Overview

The proposed approach can be viewed as a candidate element of a "defence in-depth" approach that seeks to compliment accountancy-centred activities [4]. It utilizes:

- plant design information and early consultations;
- plant process operational data/declarations;
- extensive inspector's data.

The inspector's data would largely derive from a combination of pressure and temperature measurement systems installed in vessels [5]. Appropriate data evaluation tools will be required for these systems, which represent a considerable investment in both hardware installation and software development.

The measure described here is a concept for such an evaluation tool, and is based on the concept known as *analytical redundancy* (Appendix 1). Put simply, a plant can be viewed as a number of units that are connected together: if an incident occurs in one unit, its effect is likely to be observed in others so that any local effects must correlate with effects observed elsewhere. This can be likened to a jigsaw puzzle: not only must the picture on a particular piece of jigsaw relate to the overall picture, but also the piece must connect with neighbouring pieces. Put more formally, analytical redundancy exploits the inherent static and dynamic relationships among the measured variables. In other words one makes use of a *mathematical model.* Mathematical modelling is not new to nuclear materials safeguards; for instance accountancy is based on a discrete model of the hold-up in a plant. The concept here seeks to extend *this* model, and not to propose a quantum leap in the type of model adopted. Thus the concept is based on continuous materials balances like those for a tank:

$$\frac{dM_{bulk}}{dt} = f_{in}\rho_{s_{in}} - f_{out}\rho_s \tag{1}$$

and

$$\frac{dM_{Pu}}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu] \tag{2}$$

where M is a mass, f is a volumetric flow rate, square brackets [] are used to denote a concentration and ρ_s is the solution density. Some of these flow rates would also appear in the materials balances that would be formed for the plant units that are connected directly to this tank, because transfers are made from one unit to another. The equations would be connected. By analysing these transfers, the underlying goal is to confirm that the nuclear material contained in a plant can be distributed in such a way that it correlates with an inspector's perception of what is going on in a plant, and with the measurements that are available. This goal would be assessed continually because reprocessing plants operate in time. To do this, use is made of computer simulations to continually predict the distribution of material by solving these balances; this is akin to a continuous version of a discrete simulation of the hold-up in a plant i.e. of a running book inventory. Clearly in the process of doing this other goals arise, which increase the system's capability, and these will be discussed with the detail.

Figure 1 shows the way the concept would be implemented. Key novel features include the Real-Time Database, consisting of all measured and some calculated variables at all time points, the Operational History Database, containing the system's perception of plant operation, and of course the Evaluation System where the goals would be assessed.

Such an approach is in accordance with the views of the LASCAR Working Group, which proposed that the verification of inventories of bulk materials should be supported by monitoring and evaluation of selected operating parameters.

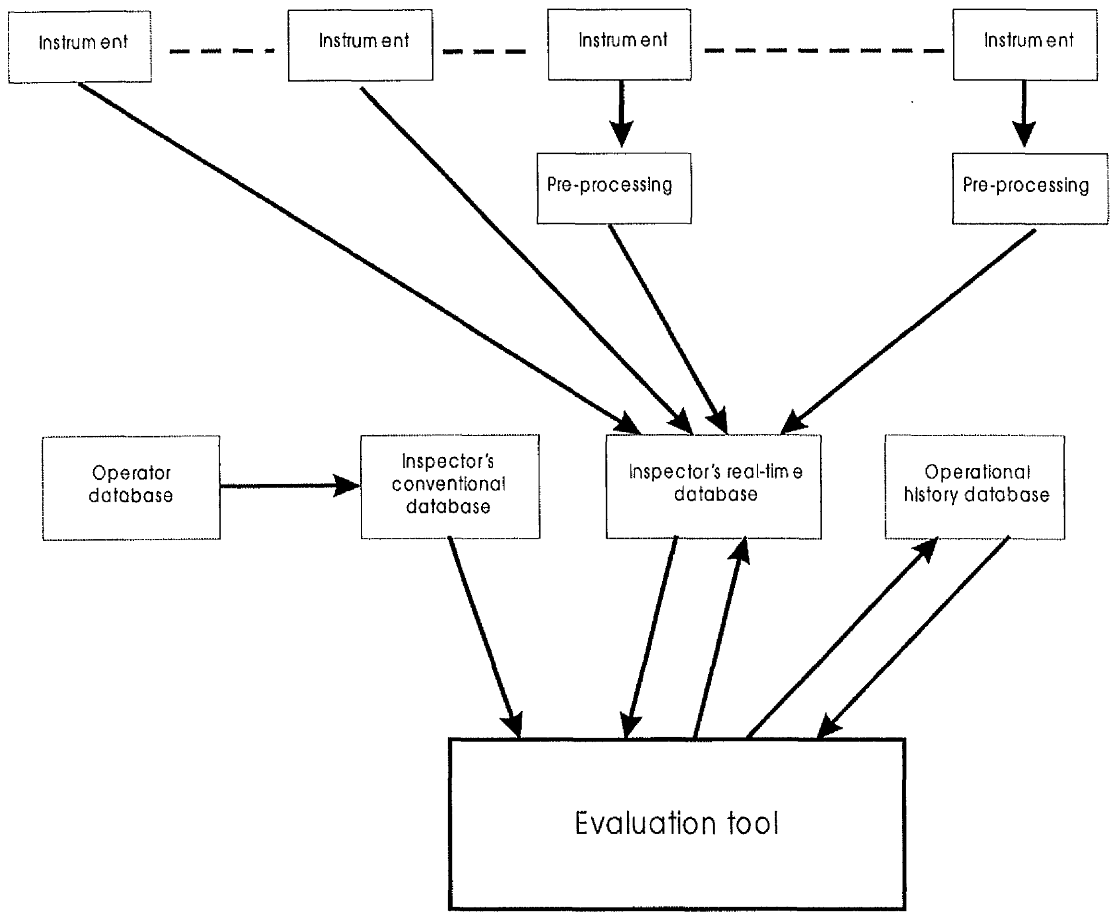


Figure 1: The Evaluation Tool's Place

2.2 Overall Aims

This report focuses on the process area, brief descriptions of possible approaches for accountancy tanks, storage areas and finishing plants are given in Appendix 2. Many operations occur daily in a process area of an operating reprocessing plant, e.g tanks are filled and emptied. These activities can be observed by looking at inspector data such as would be available through Solution Monitoring (e.g. by observing changes in tank levels).

The overall aim of this approach is to confirm that observations derived from inspector data correlate with the operations declared by the operator and that no other operations have arisen; all significant features in the observations must be explained. In practice a number of disagreements are likely to arise daily. Operators may fail to declare an operation, plant problems may lead to unusual operations, sensors may function incorrectly and so on. Most of these *events* are of little interest to a nuclear materials safeguards inspector and so a number of stages of data processing and evaluation would be required to extract those events that might be of importance:

detect disagreement	→	hypothesise possible causes	→	follow-up	→	unresolved disagreement

The main visible output from the system would be a list of events. Reasons as to why they have been generated must be given so that these alternative causes can be evaluated.

This concept for evaluation software has been developed with the following objectives:

- to minimise the number of disagreements that arise in the first place;
- to minimise the inspection effort in the detection and hypothesis stages;
- to follow-up on only those disagreements where there is serious doubt;
- to describe the disagreements in a way that is transparent to the inspector; the intention is that, on the majority of occasions, the inspector would 'accept' the disagreement as being of no further interest;
- to suggest follow-up action when needed.

The data evaluation system must be robust. Evaluation should not make too many assumptions about the data that is likely to be collected. For instance, although a transient in level observed in a certain tank might often look the same as before, one cannot assume that this will always be the case.

The methods described here would be implemented as computer software. Although commercial, high-level software tools are available that enable the systems developer to produce systems relatively easily, a considerable amount of effort would still be required in specification, programming and testing. For purposes of cost and of QA, it is important that the specification is clear and well structured. Given the need not to overburden the inspector, the user interface is of particular concern. The next sub-section outlines the various stages.

2.3.1 The various stages

Detect a disagreement: in order for the evaluation system to detect an irregularity, the system must have actual *plant data* and *reference data* with which to make a comparison. This reference data, which describes what would be expected to be observed if the plant was to be operated as expected, would largely derive from simple mathematical models of the plant. It is also important that the right plant data is collected: the system will only be able to detect an irregularity if it can be observed in the data[1]. At this stage, the source of the disagreement doesn't matter, only its effect on the observed variables.

[1] In fault detection circles the term *characteristic variable* is used to denote an appropriate variable

Hypothesise possible causes: in order for the evaluation system to generate hypotheses for why a disagreement has arisen, the system must be based on a methodology that allows as many alternatives as possible to be admissible, and to allow for the possibility that the cause has not been previously thought of. Simply saying, for instance, "Features X, Y & Z must imply Diversion D" would soon lead to false alarms. Instead it is proposed that the system offers likely alternative causes to explain the observed data, categorised as either 'normal operation', 'operational fault' or 'needs follow-up', and orders these in some way.

Follow-up: each cause that is categorised as 'needs follow-up' would have a set of instructions associated with it, which would seek to confirm that that particular cause had actually occurred. If a particular cause is identified then it is likely that the disagreement would be re-categorised as either 'normal operation' or 'operational fault'. However it is possible that it would remain *unresolved.*

Unresolved disagreement: it is important that the reasoning process that leads to this declaration should be transparent. In addition sufficient evidence must be available to justify this action.

2.4 Additional Assurance

Additional assurance would be obtained because of an increased capability to gather and to correlate evidence. An increased deterrence would be achieved by increasing the risk of detection through

- an increased awareness of tank operations; this is especially because undeclared changes in the contents of tanks would be correlated with effects observed elsewhere;
- an increased transparency in the operation of other process units;
- an increased likelihood to detect potential falsification of information through the use of corroborating instrumentation and through analytical redundancy.

Various case studies have shown that the system is capable of responding to certain diversion scenarios that are smaller than those that could be detected by an NRTA system. The term 'responding to' is used to highlight the fact that the results from the evaluation would not be in the form of alarms (i.e. in detection) but of information, layered to minimise inspector interaction.

2.5 Data Sources

2.5.1 Reference Data

Since plant data is temporal, so is reference data. Thus it is envisaged that the real-time database (see Figure 1) would also contain data fields pertaining to calculations, as opposed to measurements. The processes involved in producing suitable data also provide opportunities for detecting disagreements. It is important to appreciate that disagreement detection would be distributed in that a number of different tools would be involved. Before discussing these processes, data availability is first examined. The data might come from a number of sources: from the declared operation, from the operator's DCS, and from inspector data.

2.5.2 Declared Operation

The declared operation would be in two parts: Category A and Category B. Category A would include that data that is declared conventionally, i.e. the *operator's declaration.* The *operator's declaration* would include data pertaining to KMP transfers, an "interim" physical inventory for NRTA and the conventional physical inventory taken relatively infrequently. Category A is then parameterised by a set of times of events, temperature measurements, level measurements, sample concentrations and sample densities that pertain to these points and vessels. Operator's declaration – Category B pertains to new additional data, which primarily consists of qualitative descriptions of the operation of the main stages of the process. Very little of this data would be used in the system directly, and none is essential. Its main purpose is to avoid unnecessary operator/inspector dialogues, which might otherwise arise if a disagreement is detected. Two examples are given here to explain this.

1. The Cycles and Concentrator represent hidden inventories, the contents of which would fluctuate with operation. Although the flow sheet and hence (approximate) inventory might be deduced by monitoring the various streams, this would be somewhat intrusive and complicated. It is far easier to "confirm" a flow sheet that has been declared.
2. Depending on the quantity of solution that leaves a tank during sampling and hence on the magnitude of the level depression thus caused, and on whether or not 'acid substitution' is an issue, sampling times might or might not be of interest.

The mode and frequency of declaration is open to discussion. Essentially there would a series of asynchronous declarations like "cycle_1 flowsheet changed at time … ", which could be spawned automatically by the operator's DCS, or entered via an electronic form when it happened, or once a day, or on demand. It is important to appreciate that this information would not be essential, it would just facilitate transparency.

2.5.3 Additional Operator Data That Will Be Required

This data differs, conceptually, from the above because an arrangement would have to be made with the operator so that DCS records could be transferred to Agency databases. There would be two types: data from other in-process instrumentation, and data pertaining to process controls. The precise information used in the evaluation would presumably be specified based on discussions between the Agency and the operators. Here we assume that all such information would be made available. However it is important to stress that not all data is essential, the acquisition of the various pieces of information merely leads to a more straightforward evaluation system. This will be discussed later.

Examples of the two types are shown below.

- Neutron detectors located in the main extraction and plutonium purification cycles to detect spontaneous fission neutrons from Pu isotopes plus other neutrons from reactions in the solution; their outputs are usually compared with an alarm threshold.
- X-ray fluorescence monitors (to indicate relative atomic composition).
- Flow meters installed in the solvent-extraction area: these have low accuracy but high precision. Thus although they might not be appropriate for quantitative calculations, they could be used to detect/confirm changes in plant operation.
- Data pertaining to the concentrator heating element.

- Some form of indication that acid molarity is remaining constant at the outlets from the solvent-extraction cycles and from the concentrator (electrical conductivity?).
- The set-point of the flow metering device on each of the active feeds;
- The set-point for the plutonium concentration output from the Concentrator.

2.5.4 Inspector Data

In addition to the nuclear materials data that is usually collected, inspector data would include pressure dip-tube measurement records pertaining to tank levels and densities plus temperatures [5]. The rates at which data would be recorded would be sufficiently fast to enable appropriate data analyses to be made. Although the data acquisition system would presumably convert this data into volume/density/temperature records, both the raw and the converted data would be stored. The raw data would be used to validate operation of the dip-tube system whilst the converted data would be input into the real-time database. It is well-known that the level/density measurement system has a propensity to be self-validating because the two key dip-tube pressure signals are highly correlated. To reduce the complexity of the analysis system, and because it is sensible anyway, it would be preferable that faults internal to level/dip-tube instrumentation should be isolated separately. Adherence to the recently proposed standard on SEVA sensors might be beneficial here because this would provide a communication protocol between the sensor validation system and the evaluation system that includes an instrument health status. Thus a system is envisaged like that shown in Figure 2. Although not essential it would be useful if data collection was synchronised between tanks i.e. the 'output the average' signal was synchronised. It is also worth pointing out that it does not appear to be uncommon for tank levels to drop below their density dip-tubes. Such eventualities must be accommodated in any data evaluation software.

It is also recommended that samples should be taken from certain tanks for a number of reasons:

- to guard against 'substitution' of solution with acid;
- to validate certain assumptions made about the reference data.

There is clearly a limit to the number of samples that could be collected and analysed. An analysis might be required as part of a follow-up activity but not all samples collected need be analysed. In order to provide a certain level of effectiveness, it is recommended that samples be collected and stored, temporarily for a day or two and frequently. An alternative approach that is probably less powerful but more practicable is to adopt some form of random sampling strategy in which a sample is taken from each tank every now and then.

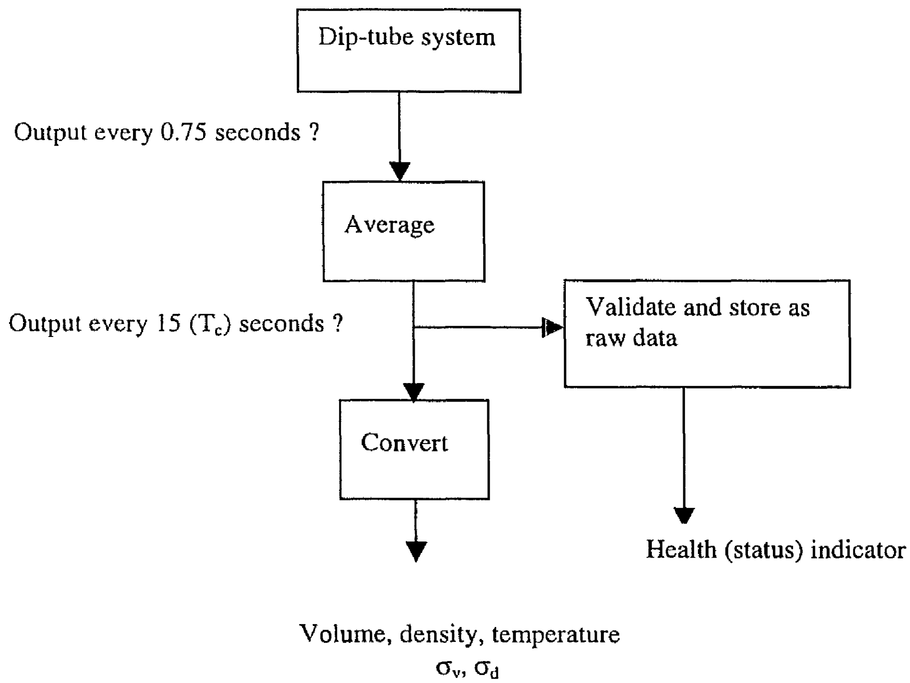


Figure 2 Dip-tube output pre-processing

2.6 Strategy

In very broad terms, the approach is based around simple simulations of the plant. Simple simulations are used in a number of different roles: in generating reference data, in generating boundary conditions for other simulations, and in reasoning about disagreements. Since the plant data is largely composed of tank volumes and densities, the focus of the simulations is on these variables. However plutonium is of primary interest, and hence there are also simulations that describe its flow and hold-up. The underlying strategy, which can also be stated in terms of a simulation, is given in the next paragraph.

A plant is deemed to be operating as declared if a simulation can be constructed that is in agreement with plant measurements over two different time intervals:

- short term: broadly equivalent to 1.5 cycles of filling/emptying a buffer tank;
- medium term: broadly equivalent to ten cycles of filling/emptying a buffer tank.

Note that the long term is not considered because other approaches would be more appropriate. Evaluation over the short-term is to provide assurance that something like 0.1 SQ is not diverted over a period ≤ 1 cycle.

Evaluation over the medium term is

1. to provide assurance that something like 0.25 SQ is not diverted over a longer period ≤ 10 cycles;
2. to obtain additional assurances by
   a) confirming that samples correlate with our understanding of plant operation;
   b) confirming that the interim PI, which is taken for purposes of NRTA, correlates with our understanding of plant operation;
3. to maintain an understanding of the way material is distributed throughout the plant. This would also provide initial conditions for the short term evaluations.

The simulation is constructed by taking a fixed model (i.e. one whose balance equations etc. are specified *apriori*) and driving it with material movements derived in a number of stages:

1. flow rate estimation;
2.
   - short term: constrained adjustment along pre-defined paths;
   - medium term: inverse modelling.

This leads to a structure in which a number of tools (modules) are engaged to build-up a 'picture' of plant operation that is 'housed' in two databases, the Real-Time Database and the Operation History Database. The Operational History Database is used to store that data that is not time-stamped. It need **not** contain a complete record of plant activities and is more likely to be based on data objects. There would be three categories: those pertaining to *declared operation*, to *events* and to *process units.*

It is important to appreciate the temporal nature of the evaluation environment, plant data would be continually entered into the real-time database whilst evaluations take place. In this environment different activities would take place at different times, co-ordinated by some kind of 'Executive', which would have processes responsible for the short and medium term activities. This environment is outlined in the next main section.

2.6.1 Plant Mathematical Models

The plant models would be dynamic, quantitative and deterministic. They would be based on the plant design, and on possible plant operation and would be substantiated through the DIV process. They would not be what are commonly known as "process models" because they would not contain detailed equations pertaining to solvent extraction or evaporation. Instead they would largely derive from balances of water, plutonium, nitric acid and any other components where the choices of matter 'balanced' would vary and depend on the process units involved.

2.7 Complexity and Inspector Involvement

By now the reader might be getting an impression that the system proposed will be complicated and involve considerable inspector effort. As with nuclear materials accountancy systems, there are compromises: inspector involvement versus software complexity, power to detect versus false alarm rate and so on. The aim here is to minimise inspector involvement by automating as much as possible. The resultant software complexity would then be handled, in part, by selecting appropriate software tools, in part by aiming for a professional, software engineered implementation of modular construction and in part by ensuring that the implementation is not overly dependent on any one evaluation tool. One major aim must be to ensure that there is a proper software specification. Another must be to ensure that the system would remain operational, at least sub-optimally, even if a hypothetical situation arises where all the evaluation tools prove to be totally useless.

However, complete automation is undesirable because some degree of user interaction is sensible. As with nuclear materials accountancy systems, the aim is to detect the occurrence of certain situations without having to respond to other situations (i.e. false alarms) as well. The difficulty is in discriminating between these two classes. Although the system could be configured so that it only detects really serious situations, this would raise two concerns: firstly, given that the system might never alarm, how does one know that it is working properly?, would anyone have the expertise to access it if this very unusual event actually occurred?, and secondly, how does one define a 'really serious situation' anyway? Thus there is a need to interact with two types of person: with an inspector, largely for training/refresher purposes, and with someone from systems studies, who is able to monitor and to adjust (i.e. to evolve) the performance of the system.

2.8 Data Collection and Storage

It is important to have a high data collection rate (e.g. every 15 seconds) to view the start and stop of process operations clearly. If data storage is an issue then frequent archiving is preferable to data compression. If data compression is to be used, then algorithms should be chosen with care so that pertinent features are observed [6]. Similarly if the hardware form of data compression, i.e. multiplexing using for instance scanivalves, is to be used then their outputs should be recorded with care. As part of the follow-up activities, sometimes it might be necessary to discuss a particular irregularity with the operators. This might lead to the operators consulting their own records to corroborate or refute the disagreement. The most appropriate measurement records are likely to contain measurements pertaining to tank volumes and densities. If data compression is used in the operator's data recording system and it is different from that used in the inspector's, then discrepancies might well arise. This issue should be addressed before a detailed systems specification is produced.

3. THE EVALUATION ENVIRONMENT

The composition of the evaluation environment can be described in a number of different ways. From the function viewpoint the environment can be described in terms of an Executive that invokes *tools* to process & evaluate the data; the aim is to confirm that the plant is operating as specified. From the implementation viewpoint the environment can be described in terms of computational processes that interact with data input and with data stored in databases; the aim is to generate a list of events that describe any disagreements that have been detected and not explained. This section outlines the function viewpoint, the implementation viewpoint is discussed in the next section. A number of examples are given in Appendix 6 of how the various tools combine together to evaluate a particular scenario.

3.1 Plant Topology

For simplicity it is assumed that all the plutonium takes a 'single' route through the plant. The plant can then be viewed as a sequence of segments each composed of either a tank set or a process area. Where branching does occur, for example for reworking and at Pu/U separation, plutonium flows along the secondary branches would normally be viewed as flows to hidden inventory unless the actual inventories of these branches are estimated. If, for instance, appropriate data is available for a rework tank and the rework route spans two segments, then the rework tank would be included in the exporting segment.

3.2 The Executive

The role of the Executive (Appendix 3) is to confirm that the plant is operating as specified. It does this by invoking various tools, either when appropriate data becomes available or when a particular time is reached. The tools are contained in toolboxes, which address different functions (Table 1). Toolboxes 1-4 contain tools that provide those boundary conditions needed by the simulations. Toolboxes 3&4 have another role in that they contain tools that estimate flow rates, which are good indicators of plant operation, and hence can be used to detect disagreements (Toolbox 6). Toolbox 7 contains tools that provide hypotheses to explain the disagreements and then to output the most appropriate as *events*. Toolbox 8 contains tools, which gather data to corroborate these hypotheses.

The Executive is therefore *driven* by *signals* (which announce that a particular data package has arrived) and by *timers*. Its main output is that of an *Event List*, which contains events layered in order of importance. Figures 3-6 show the order in which the Executive schedules the tools: Figure 3 pertains to a tank-set, Figure 4 to a solvent-extraction cycle, Figure 5 to the concentrator and Figure 6 to a rework line. Measurement data flows from the tanks, solvent-extraction cycles and concentrator, both into databases and also into evaluation tools. The Executive schedules on the basis of *decision boxes* and boxes marked *short-term assurances* and *medium-term assurances*. For instance, box 'Decision 1' executes Tool 3b once data pertaining to a single cycle has been collected. Box 'Decision 2' schedules execution of Tool 5, configured explicitly for a particular tank set: it is invoked every time the tank set is deemed to have completed an operational cycle, which is usually the same as 'when the buffer tank has completed an operational cycle'. (A timer is also provided to ensure that Tool 5 is always invoked at least once every 12 hours). Short-term assurances are only sought in response to an output from either of the detectors (Tools 6a & 6b). Medium term assurances are likely to be sought once daily. Note that assurances for the solvent-extraction stages and concentrator are not obtained explicitly, but implicitly by evaluating their interactions with neighbouring tank sets. Short-term and medium-term assurances are derived through what is known as model-based reasoning.

Toolbox 1:	Estimation of transfers into/out of a buffer tank.
Toolbox 2:	Generates nominal plutonium inventories for the solvent-extraction cycles and for the concentrator
Toolbox 3:	Estimation of bulk flow rate – receiving/feeding tanks only.
Toolbox 4:	Estimation of the X and acid components in the continuous stream into a receiving tank.
Toolbox 5:	Plant simulations.
Toolbox 6:	Disagreement detection.
Toolbox 7:	Model-based reasoning.
Toolbox 8:	Confirmation of Operational Unit Statuses

Table 1: The Toolboxes

3.3 Events

Events are those instances that are deemed to be abnormal in some way. The operator can input these instances as part of the declaration (e.g. addition of acid), or they can be inferred by the evaluation system. In practice all events will be of a temporal nature (i.e. they will have start and stop times) and their effects might continue after their stop times and be observed by more than one evaluation tool. For instance material might disappear into hidden inventory at one time and re-appear at another. Because of this, each event is represented as a set of *sub-events*, each element of which focuses on a particular aspect that can be described by its *symptoms*. *Diagnoses*, to explain the symptoms, are therefore attached to the sub-events.

3.4 Model-Based Reasoning (data reconciliation)

Model-based reasoning is used to hypothesise sets of explanations (events), which, when applied to the computer simulations, predict a distribution of material that correlates with the declared operation. It aims to present reasonable explanations in terms of

- faulty instrumentation, transfers between monitored tanks, tank calibration equations and so on;
- additions of material;
- withdrawals of material.

Here model based reasoning draws on both quantitative and qualitative techniques. The quantitative techniques are based on various approaches to what is known as 'inverse modelling'. The qualitative techniques are based on rules (known as *productions*) and search strategies. The overall philosophy is one of evidence gathering and hypothesis generation rather than of finding 'the explanation' for a particular disagreement. This is because the evaluation system is likely to have incomplete knowledge and the data that is available is unlikely to indicate a unique 'explanation'. The hypotheses would be ordered with the most appropriate selected automatically. Each hypothesis would be supported by *symptoms* and *diagnoses*.

To explain the underlying argument that is involved, consider a sequence of N plant units (tanks, cycles and concentrator) and construct a plutonium mass balance for each unit. If all the right hand sides of these equations are now summed together, and then all the left hand sides are summed together, one obtains:

$$\sum_{1}^{N} \mathrm{Mass}_i(t) = \sum_{1}^{N} \mathrm{Mass}_i(0) + \int_{0}^{t} \mathrm{flow\_rate\_in}_1(t) - \mathrm{flow\_rate\_out}_N(t)\, dt \qquad (3)$$

That is, and as would be expected, the change in total mass is governed solely by the flow rates, or transfers, at the start and finish of the sequence. If the sequence represents an MBA then these transfers are at KMPs, which are already monitored accurately; thus the total mass is already monitored and the internal flow rates merely determine how the material is distributed. This distribution cannot be measured directly, but must be inferred from what can be measured and from knowledge of plant operation, which is encapsulated in the computer simulation of the plant. Of key importance is the fact that the operation of the tank sets are observable, directly, whilst the operation of process units are not. This has a number of implications: firstly each non-observable unit can be viewed as a single entity or unit, secondly, if a disagreement in an observable unit can be 'explained' by a material transfer across its external boundary, this material can hide, temporarily, in a connecting non-observable unit. This results in three stages of analysis that seek to answer the following:

1. how can the material be re-distributed to satisfy that observed?
2. can this re-distribution be corroborated by supporting evidence?
3. what event description best matches these re-distributions?

Re-distribution is examined in two different time-frames, the short term and the medium term. The system is not intended to be used to detect incidents that span the long-term (e.g. a diversion of less than say 0.005 significant quantity per hour). Although nominally model-based reasoning is assigned to Toolbox 7, it also makes use of simulations (Toolbox 5) and detectors (Toolbox 6). The reasoning process is therefore co-ordinated by the Executive and Toolbox 7 contains tools to support the reasoning processes.

3.5 Compensating For Tank Calibration Errors

Being based on flowrates estimated on the basis of volume changes, the approach is particularly sensitive to errors in volume change estimation. Additive errors in tank calibration are not of major concern because the emphasis is on volume changes and additive errors cancel out when calculating these changes. However, as explained by Burr & Wangen [10], multiplicative errors do not cancel out. A number of test cases are shown in Appendix 7, which have been produced to show how multiplicative errors affect the redistribution of material in a simulation as compared to in the real plant. The way in which material would redistribute over time would depend on the distribution of the multiplicative errors and on which volume data is used to estimate which flowrate. An error in a tank does not result in a redistribution at all (i.e. the effects of the error are confined to the tank itself) if certain flow rates are used in the calculation. However redistribution does occur otherwise especially if a number of tanks have significant multiplicative errors. Clearly the effects of these errors would tend to mask the visibility of other activities going on in the plant and hence tools are needed to identify and accommodate their existence. Toolbox 7 therefore contains two tools (7b-4 & 7b-5) that can be used to identify these errors.

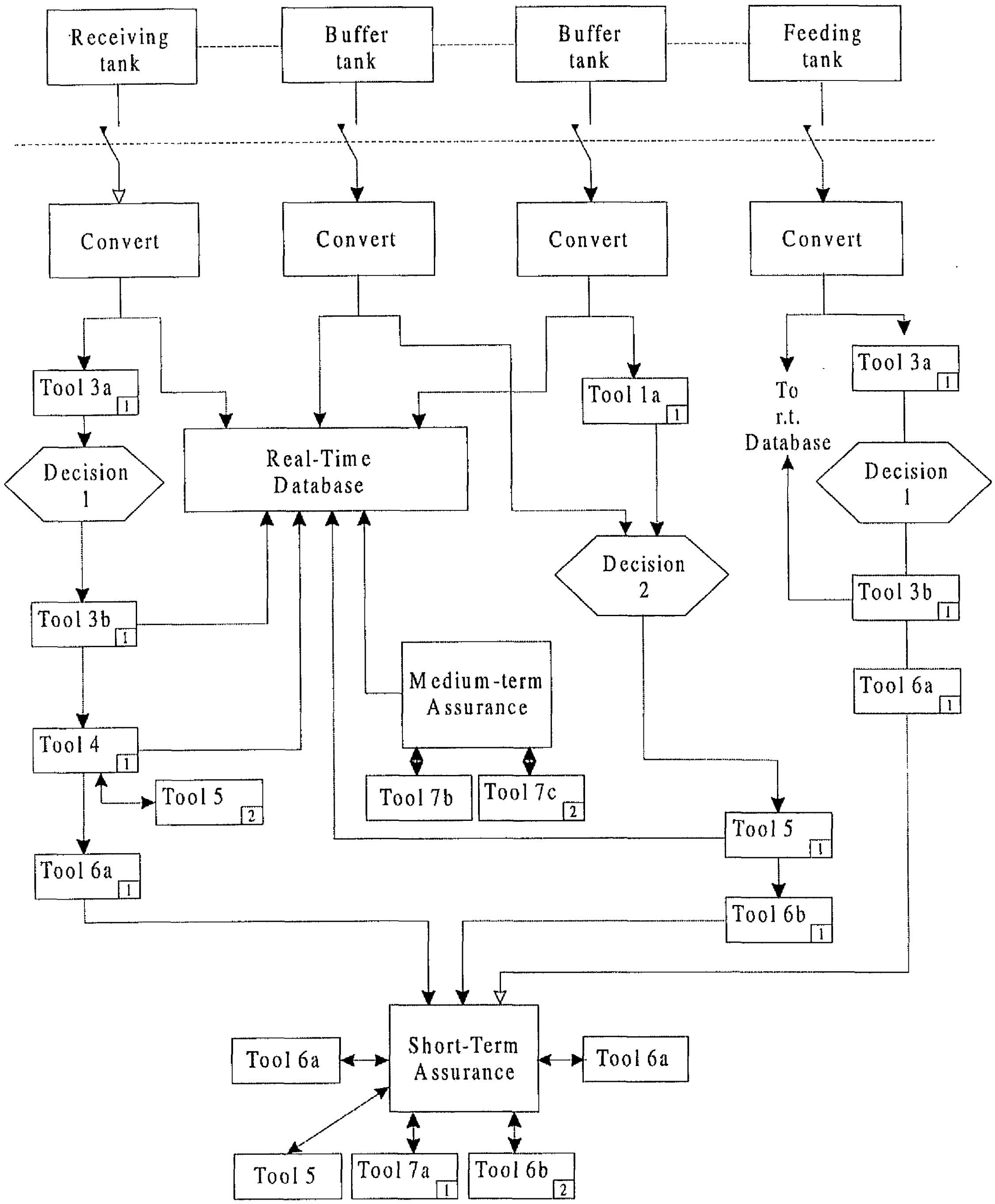


Figure 3: Tool invokation schedule for a tank-set

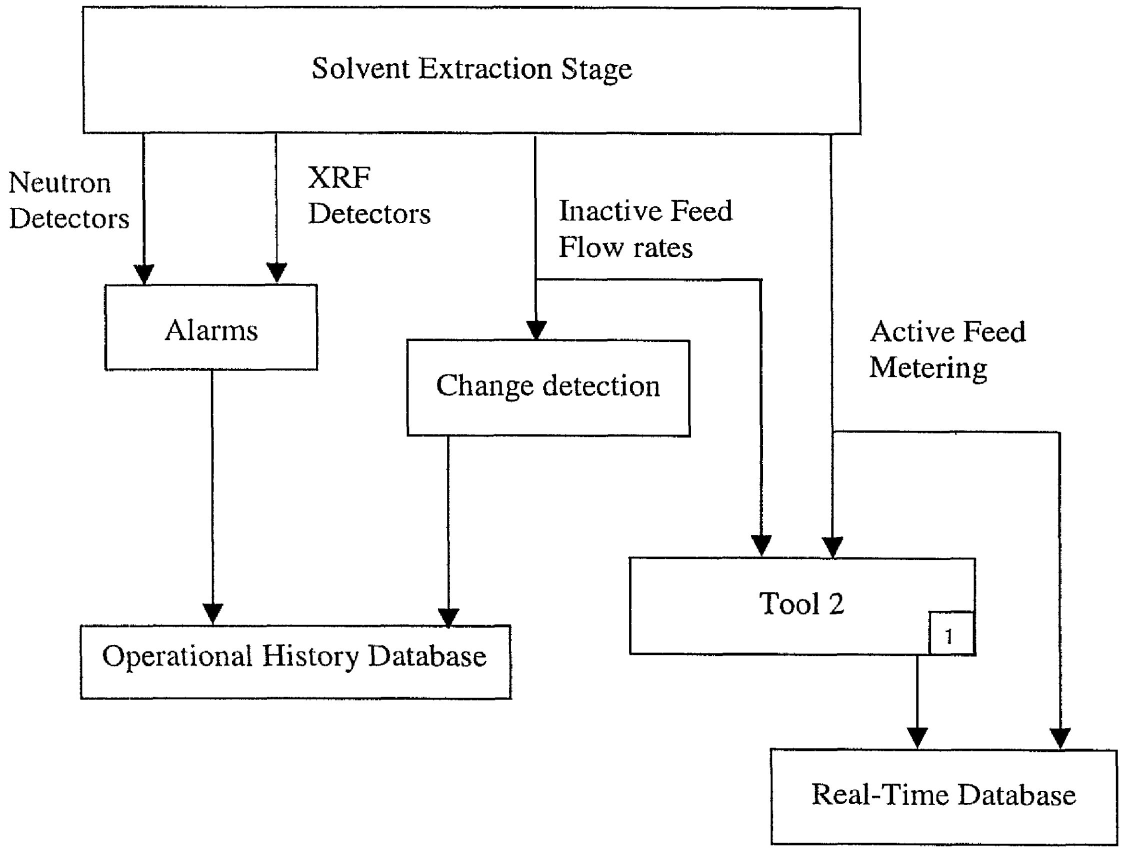


Figure 4: Data flows for a solvent-extraction cycle

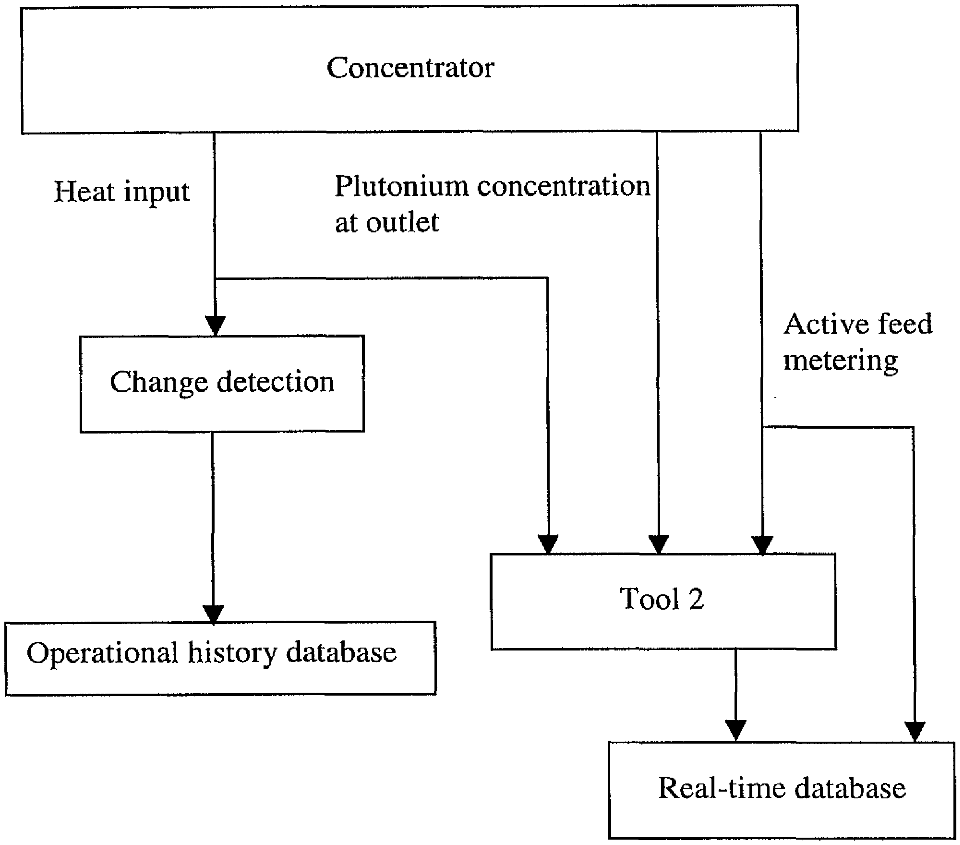


Figure 5: Data flows for the concentrator

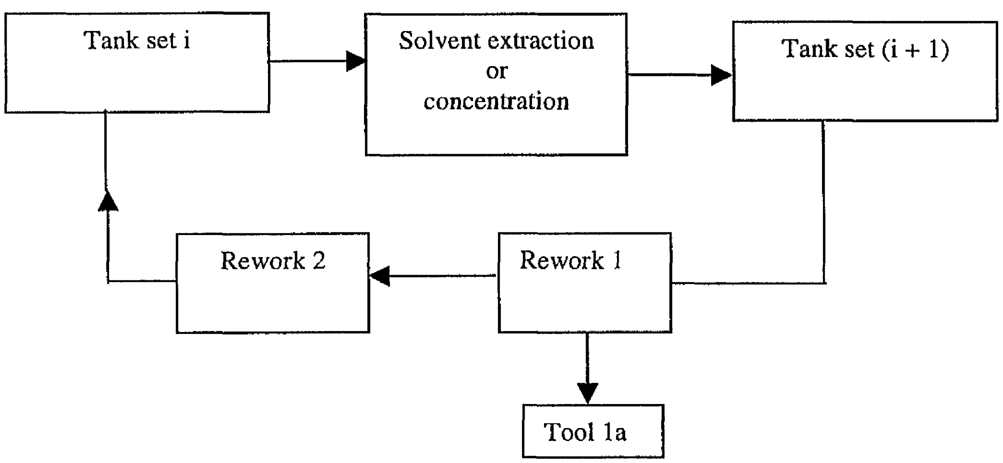


Figure 6: Data flows for re-work tanks

3.6 Density measurements

Before describing the individual tools, it is important to appreciate that density measurements are just that. At a given temperature T, the density, ρ_s, of any solution, which is likely to be present, can be calculated on the basis of a correlation like [8]:

$$\rho_s(T) = \rho_w(T) + \alpha_{Pu}(T)*[Pu] + \alpha_{H+}(T)*[H+]+..... \qquad (4)$$

where $\rho_w(T)$ is the density of water, [Pu] is the volumetric concentration (g/l) of plutonium, [H] is the molarity and $\alpha_{Pu}(T)$ & $\alpha_{H+}(T)$ are coefficients. If the solution density, acid molarity and temperature are known then the concentration can be obtained. However if only a density measurement is available, and without making any further assumptions, the individual components cannot be estimated separately. The term X is used to denote this fact:

$$\rho_s(T) = X + \rho_w(T) \qquad (5)$$

The incorporation of density measurements therefore depends on the assumptions that can be made about the composition. Since the simulation starts at the input accountancy tank, where composition is known, and works forwards through the plant, individual components can be traced as far as the first 'hidden inventory'. This can be achieved by adding mass balances pertaining to acid molarity, U and unsp (unspecified: for all other components) to the simulation. Plutonium concentration can now be confirmed using the density measurements. A similar process can be applied after the first cycle, and so on, provided that the composition can be assumed at the start of each set of tanks. Tools in Toolbox 4 4 is intended to obtain the information required.

3.7 The Individual Tools

Each tool is outlined here; a more detailed description is given in Appendix 5. The introduction to Appendix 5 also contains descriptions of two methods that are used extensively by the tools: Cusum detection and recursive least squares (RLS). These tools read from and write to the Real-Time Database: for ease of reference, Appendix 4 tabulates some of its fields.

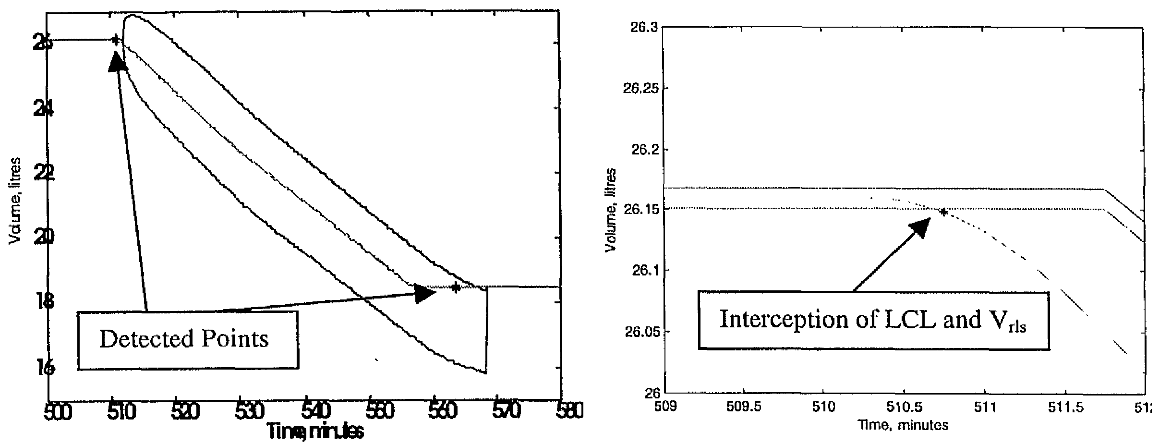
<u>Toolbox 1: Estimation of bulk transfers into/out of a buffer tank.</u>

Toolbox 1 contains two tools, Tool 1a is for normal batch tanks whilst Tool 1b is for accountancy tanks. The description here is of Tool 1a. Estimation would be based on changes in tank volume. This is straightforward if the transfer is to or from another buffer tank as opposed to a feeding/receiving tank. It is also straightforward if the time taken to transfer to/from a feeding/receiving tank is <u>always</u> relatively small. Note the emphasis is on the word 'always' since this is up to the operators. If this is not the case then a more sophisticated approach is needed (Toolbox 3).

The automatic estimation of a volume change might not be that straightforward: noise might be present on the volume measurements and the transfer might 'taper' asymptotically (i.e. take a long time to finish). Hence it is unlikely that the most appropriate algorithm can be specified without reference to actual data. The method proposed here was developed with reference to TRP batch transfer data [14] and should be close to what is required. It combines the concept of the standard Shewhart control chart [22, 23] with RLS estimation. An RLS estimate of volume, V, is produced (i.e. V_{rls}) to filter the 'noise'. This estimate is delayed slightly to improve the detection of abrupt changes and the errors ($V - V_{rls}$) are then tested against lower and upper control limits:

$$\text{Upper Control Limit: } V - V_{rls} > F_U\sigma_x \qquad (6)$$
$$\text{Lower Control Limit } V - V_{rls} < F_L\sigma_x \qquad (7)$$

where nominally σ_x is the standard deviation of ($V - V_{rls}$) , and F_U & F_L specify the maximum allowable deviation of the data. A change is detected when V intercepts either control limit. Figures 7a & 7b show the algorithm detecting the start and stop points of a transfer out: the centre line, V_{rls}, is enveloped by upper and lower control lines, which increase significantly when a change occurs. Both the points identified in Figure 7a have been obtained by first detecting a change, selecting the next data point in time (as in Figure 7b), then working backwards through the time series, V, to find the most likely turning point.



a) b)

Figure 7: a) V_{rls} enveloped by upper and lower control limits during a typical transfer with points detected as shown; b) close-up of start of transfer showing data point that is chosen prior to tracing backwards

<u>Toolbox 2: Generates nominal plutonium inventories for the solvent-extraction cycles and for the concentrator.</u>

The authors have been told [9] that tools would be provided that produce these inventories on the basis of the declared operation.

<u>Toolbox 3: Estimation of bulk flow rate - receiving/feeding tanks only.</u>

An estimate of the continuous flow rate into a tank (for a receiving tank) or out of a tank (for a feeding tank) would be obtained by observing changes in its volume. This method would work in tandem with Tool 1, which would account for those other changes in volume, in the same tank, caused by transfers to/from buffer tanks. If metering device outputs are available on the continuous flow lines, this data could be used instead. Tool 3, and in the medium term Tool 7b, outputs would then provide assurances about the accuracy of this data.

Toolbox 3 contains two tools, which, when combined together, estimate flow rate. Tool 3a identifies the individual fill/empty cycles in each tank. Tool 3b analyses each cycle after it has been identified. By definition, Tool 3a is applied continuously to the incoming data, whilst Tool 3b is invoked by the Executive (Decision 1) once a cycle. A separate copy of a. would be executing all the time for every receiving/feeding tank. Thus in operation several copies of Tool 3 would be running concurrently. The overview here will be for a receiving tank. The equations for a feeding tank would be very similar.

Let the volume in the tank, at any time t, be V, the flow rate in be f_{in} and out be f_{out}. Then

$$\frac{dV}{dt} = f_{in} - f_{out} \tag{8}$$

Over a single cycle it is normal for the flow rate into the tank to remain fairly constant so that, without loss of generality,

$$f_{in} = \alpha_i + \Delta f_{in}(t) \tag{9}$$

where α_i is a constant over cycle i. The output is somewhat different:

$$f_{out} = \begin{cases} 0 & ;\ t \text{ otherwise} \\ g_{out}(t) & ;\ t{:}\ t_{s,i} < t < t_{f,i} \end{cases} \tag{10}$$

where $t_{s,i}$ and $t_{f,i}$ are the times that the i^{th} transfer starts and finishes. The purpose of Tool 3 is to estimate f_{in}; out flow rate g_{out} is estimated as a bonus. There are four stages:

Tool 3a: identification of a single cycle (i.e. fill then empty) and hence of times $t_{s,i}$, $t_{f,i}$;

Tool 3b:

1. estimation of α_i;
2. estimation of g_{out};
3. estimation of Δf_{in}.

Single Cycle Identification

Parameter α_i is only estimated whilst f_{out} is assumed to be zero. Thus the key points in the cycle are $t_{f,i-1}$ (some time after output has stopped), $t_{s,i}$ (the time at which output starts) and $t_{f,i}$ (Figure 8). In practice it is difficult to detect $t_{s,i}$ and a time slightly later, $t^{+}_{s,i}$, is identified instead. Point identification is via RLS and the Cusum test.

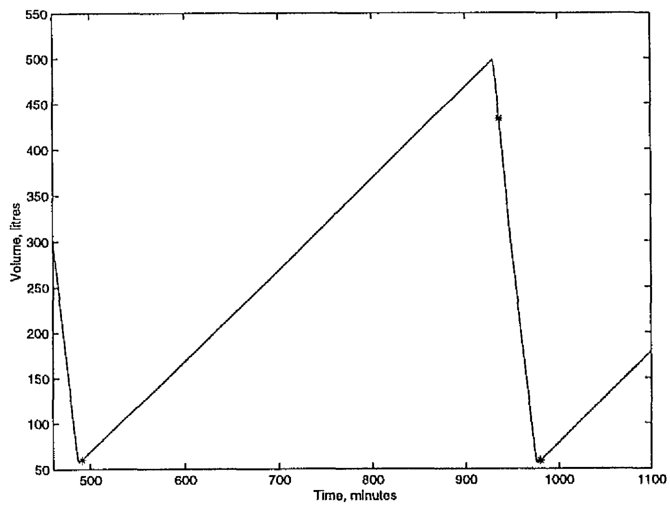


Figure 8: Fill/empty cycle

Estimation of α_i

The aim is to estimate parameter α_i over the time period t: $t_{f,i-1} \le t < t_{s,i}$, where time $t_{s,i}$ is not known. There are various ways of doing this: the crudest is to perform RLS estimation over a time period, starting sometime after $t_{f,i-1}$ and finishing sometime before $t^+_{s,i}$, when it is fairly certain that there isn't any output taking place; more sophisticated versions avoid making this assumption by estimating parameters α_i , $t_{s,i}$, $t^+_{s,i}$ and β over the time period t: $t_{f,i-1} \le t < t^+_{s,i}$, where

$$\beta = \frac{\int_{t_{s,i}}^{t^+_{s,i}} g_{out} dt}{\left(t^+_{s,i} - t_{s,i}\right)} \qquad (11)$$

i.e. an average flow rate out, β, is assumed for the very small time period t: $t_{s,i} \le t < t^+_{s,i}$. In the latter, Equation 7 is solved to provide volume estimates V, which can be compared with the measurements taken from the tank. Estimation is then achieved by choosing those parameters that minimise the cumulative sum of the differences squared:

$$Errsum = \sum_{i=1}^{m} \left(V_{sim_i} - V_{meas_i}\right)^2 \qquad (12)$$

where:

$V = tank\ \ volume$

$meas - measurement\ value \quad\quad sim - simulated\ value$

Simulated annealing is used to identify these parameters.

Estimation of Flow out

Numerical differentiation is performed over the time interval t: $t_{s,i} \le t < t_{f,i}$. Alternatively the flow rate can be obtained by looking at the tank downstream.

Estimation of Δf_{in}

This flow out estimation is then used in the estimation of Δf_{in} over the time interval $t: t_{f,i-1} \leq t < t_{s,i}$. Even during normal operation it is likely that the flow into/out of a feeding/receiving tank will deviate significantly from α_i. An observer is used as the estimator (see Toolbox 7).

Toolbox 4: Estimation of the X and acid components in the continuous stream into a receiving tank.

Toolbox 4 contains a single tool: an estimate of the concentration of X flowing into a receiving tank would be obtained by observing changes in its level & density. The acid concentration would then be inferred from equation 3 because the plutonium concentration would have been obtained from the simulation and constant (near zero?) values could be assumed for all the other components.

Toolbox 5: Plant simulations.

The methodology uses simulations in many different guises. Toolbox 5 encompasses those that are what one might call 'straight forward simulations'. Starting from the input accountancy tank and provided that estimates of the inter-tank volume transfers are available, it is straightforward to simulate the flow of material through the plant until it reaches the first 'hidden inventory' (a solvent-extraction cycle). This cycle would be modelled as a 'black box', where the steady state plutonium inventory would be available from another source (Tool 2). The plutonium/uranium split would also be mirrored in the simulation. On the uranium side, the simulation would stop once the uranium has passed through the receiving tank and one buffer tank. On the plutonium side the simulation would be more extensive. The simulation would continue on through the next set of tanks and so on until the concentrator is reached. As with the cycles, the concentrator could largely be viewed as a 'black box', where the steady state plutonium inventory would be available from another source (Tool 2).

For computational simplicity the process area should not be simulated as a whole, but instead each tank set and process unit should be simulated separately as individual segments. Each segment would be composed of one or more discrete units with zero-inventory connections. If it is anticipated that there might 'normally' be shipper/receiver differences between two tanks caused by pipe 'hold-up', then simple hold-up models should be included in the simulation to accommodate small amounts of carry-over. At present it is envisaged that Toolbox 5 would contain a single tool, which would be called with an argument that identifies the segment to be simulated. The implementers might prefer to have tools that simulate each segment separately.

There are therefore two types of model, one that pertain to tanks and one that pertains to 'hidden' inventories like solvent-extraction cycles and the concentrator. The tank simulation is composed of a number of materials balances like Equation 2 i.e.

$$\frac{dM_{Pu}}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu] \qquad (13)$$

The solvent-extraction cycle model is based on the steady state follower approach. This involves the formation of pairs of equations, one to accommodate changes in the process flow sheet and one to accommodate deviations from the nominal value. The first pair models the plutonium inventory:

$$\tau\frac{dI}{dt}+I=I_{nom} \qquad (14)$$

$$\frac{d(I+I_{hinv})}{dt}=f_{in}[Pu]_{in}-f_{out}[Pu]_{out} \qquad (15)$$

where I_{nom} is the nominal plutonium hold-up of the cycle, I is its actual 'current value', τ is a time constant obtained during commissioning (Need only be known approximately) and I_{hinv} is a second inventory that models the deviations. A second pair models the uranium inventory. If it is assumed that the plant is operating to flow sheet,

$$\frac{dI_{hinv}}{dt}=0 \quad \Rightarrow [Pu]_{out}=\frac{f_{in}[Pu]_{in}-\frac{dI}{dt}}{f_{out}}; \qquad (16)$$

That is the flow rate of plutonium leaving the cycle could be calculated provided that the bulk flow rate out is both known and compatible with the flow sheet (i.e. equations 14 & 16 need to be solved; equations 14 & 15 would need to be solved otherwise). A similar equation holds for the uranium. In the absence of a more sophisticated estimator [7], a similar approach could be adopted for the concentrator.

Thus all that are required are the inter-unit flow rates, which is the function of Tools 1 & 3, and nominal hold-ups, which is the function of Tool 2.

Toolbox 6: Disagreement detection.

Detectors are used:

a) to identify disagreements in single data streams: their general location and approximate time span (Tool 6a);
b) to detect disagreements between simulation and plant measurements: start time, location (usually open-ended i.e. no stop time – Tool 6b);
c) to detect disagreements between data streams (Tool 6c).

The way in which an incident would be detected would depend on its location and on when it occurred in the relevant operational cycles. Table 2 indicates most of those possibilities associated with tanks. Thus these possibilities would be detected using either a) or b) above. Item c) would be used when looking at, for instance the change in the U/Pu ratio across Cycle 1. Figures A.6 & A.7 in Case 1, Appendix 6 shows a typical output that one might expect from a detector. Detection is deemed to be positive if the test variable deviates from zero excessively.

Having detected a disagreement, it is important that this fact is recorded so that a system response can be generated. A large part of Tool 6 is therefore devoted to the generation of appropriate data objects: on detection a sub-event is created and the reasons for its creation are attached as symptoms.

Location : Time	a)	b)
Receiving tank: whilst only receiving	flow rate in	error upstream
Receiving tank to a buffer tank: during a transfer		buffer tank level predicted high
Buffer Tank: no operations		buffer tank level predicted high
Buffer tank to a buffer tank: during a transfer		buffer tank level predicted high
Buffer tank to a feeding tank: during a transfer		feeding tank level predicted low
Feeding tank: whilst only feeding	flow rate out	error downstream

Table 2: Application of detector categories a) & b) to tanks

Toolbox 7: Model-Based Reasoning (data reconciliation)

Toolbox 7 contains tools to support the reasoning processes, which are co-ordinated by the Executive. The Toolbox is divided into three compartments: 7a for the short term, 7b for the medium term, and 7c for corroboration and event generation. The main outputs from Tools 7a and 7b are diagnoses that explain the symptoms attached to the sub-events created by the detectors.

Compartment 'a'

As written, Compartment 'a' contains a single tool, Tool 7a, which is based on a number of observers that examine different ways of redistributing material, one scenario per observer. An implementation that specifies a separate tool for each observer (Tool 7a-1 etc.) is equally acceptable. Observers [12] are a particular kind of focused simulation in which a model prediction is 'driven' (in time) towards a specified variable (i.e. a target) by manipulating another specified variable. Here they focus on a single plant component. Manipulated variables pertain to concentrations or flow rates, and depend on the plant component involved. When certain disagreements are detected, the Executive invokes each observer in turn (Tool 7a) and tests (Tool 6a) are applied to determine whether this manipulation would explain the disagreement. Time histories of the key variables and a score, which quantifies the relative importance/likelihood of that particular scenario, are then output as *'diagnoses of sub-events'*. Case 1 in Appendix 6 is an example of what might be observed. Here three observers are used to examine three routes that might explain disagreements between the simulation and both the volume and density measurements in a receiving tank. The first observer involves movement to hidden inventory, the second manipulation of both plutonium concentration and molarity in the input stream, and the third plutonium concentration and 'unspecified'.

Compartment 'b'

Evaluation over the medium term is different because here the focus is on the development of a gradual mismatch between prediction and plant data. In reality there will always be mismatches even if the plant is operating normally. This is primarily because of the difficulties in estimating flow rates accurately with the net effect that the simulation will always gradually distribute material differently to the real plant. To correct for this, and at the same time, to be in a position to detect and diagnose an event that occurs gradually, the entire plant is analysed as one. That is the focus is macroscopic rather than microscopic. Simulations are now based solely on plutonium balances to minimise the complexity of the problem and Tool 7b is designed to redistribute plutonium over time to achieve an agreement. If an agreement can only be obtained by a large, contained redistribution, then this is marked as an issue in need of investigation.

An example of a medium term evaluation is given as Case 4 in Appendix 6. Here approximately 0.005 SQ/hr is taken from a buffer tank. A disagreement between simulation and plant data slowly builds up in the product accountancy tank until, after perhaps two or three days, it is eventually detected and the problem located to the buffer tank.

Compartment 'c'

Compartment 'c' contains a number of rules sets and search strategies, which enable it to perform many different functions. Based on an 'expert system' approach, it is flexible and can evolve with operational experience and when new instrumentation becomes available. The functions that are envisaged at present include corroboration, correlation in time and temporal reasoning. The compartment is accessed via procedures, Tools 7c-1 and 7c-2. Tool 7c-1 executes most of the functions contained in the compartment, Tool 7c-2 executes that sub-set that pertains to temporal reasoning.

Tool 7c-1

Diagnoses pertaining to individual sub-events would be corroborated by examining the statuses of the process units involved. This information would be obtained by Tool 8. Sub-events would be correlated in time to see if they pertain to the same event, and event objects would be generated for the most appropriate by examining their *scores*. Temporal reasoning would also be applied to combine events in time.

Tool7c-2

Temporal reasoning would be applied to combine events in time.

Toolbox 8: Confirmation of Operational Unit Statuses

The operational statuses of the various process units can be monitored to see if they are steady or fluctuating. Although pulsed columns can exhibit transient/erratic behaviour, this is of little interest because only the receiving tank is observed, and here such effects would not normally be seen. Thus the focus is on major changes that affect, for instance, neutron detectors or the flow rate of the product out of a cycle. Clearly the extent that this data can be used depends on what is made available. There would be two types of data-stream: Boolean, pertaining to operational alarms, and analogue, pertaining, for instance, to inactive feeds. Boolean signals can be written directly into the Real-Time Database, whilst analogue signals need to be pre-processed using Tool 8 so that either I (increasing), S (steady) or D (decreasing) is recorded.

3.8 Specification of Follow-up Actions

Follow-up actions are attached to event descriptions. A particular list of actions would be read when its associated event was written to the Operational History Database. This list might seek more information, which, on writing to the various databases, might initiate more evaluations and so on. For instance a follow-up action might involve the taking of samples, the reporting of which would trigger a re- evaluation based on their analyses.

3.9 To Conclude

The functional description has described how the evaluation would focus on the flow of material through tanks sets separated by 'hidden inventories', cycles and the concentrator. In addition there would be re-work tanks. Key features that distinguish the various sets of tanks include:

- a significant difference in plutonium concentration between them;
- the liquor in the first set is made up of many components;
- a greater variation in acid molarity in the tanks after the concentrator.

4. IMPLEMENTATION

Any implementation would centre around two databases: the Real-Time Database, consisting of all measured and some calculated variables at all time points, and the Operational History Database, containing the system's perception of plant operation and providing a means of storing data in another form than that with time stamping. Data would also be available in conventional databases (figure 1) such as a database containing information about samples. Both the evaluation system and user interfaces would then interact with these databases. Although each of the evaluation tools can be implemented separately, with appropriate graphical interfaces provided, it is assumed that minimal inspector interaction is required so it is envisaged that some form of real-time software process, an *Executive*, would schedule the tools with minimal 'visible' outputs.

The purpose of this section is to repeat aspects of the previous section but from an implementation point of view.

4.1 Real-Time Data Collection and Analysis

Volume, density and temperature measurements would be received on a regular basis and written into the Real-Time Database. Software processes (Tools 1 & 3a) would operate continuously with the database. Tool 1 would estimate the batch (volume) transfers into and out of the various tanks, whilst Tool 3a would identify when a fill/empty cycle has been completed. Data pertaining to batch transfers would be entered into both the Real-Time Database and the Operational History Database, whilst data pertaining to a continuous feed/receive cycle would be entered into the Operational History Database.

Where flow metering data is available, these would also be entered into the Real-Time Database as would the Boolean outputs (alarm or not) from neutron and XRF detectors. Real-time data, pertaining to the concentrator heating element and the flow rates of inactive feeds, would be analysed in real-time and only the direction of a significant change (i.e. up or down) would be entered into the Real-Time Database (Tool 8).

4.2 Events

An event can be represented by a class of data object that has a description and set of sub-events associated with it. Each sub-event would have a set of sub-classes representing its possible diagnoses and symptoms. For instance a brief description, location, times at which the event started and stopped, its time history and so on can be stored in a database. Two examples of how the data object might be constructed are given in Appendix 8.

4.3 The Executive

The Executive is a software device that would co-ordinate the evaluation by executing *scripts* in response to *signals* that tell it that certain activities have been completed (See Appendix 3). The main signal/script combinations are listed below.

1. Detection of the end of a fill/empty cycle would trigger the execution of a script that would invoke Tool 3b (the other part of Tool 3), followed by Tool 4, which calls Tool 5 solely for the process unit upstream, followed by Tool 6a. Data pertaining to the continuous transfers (i.e. outputs from Tool 3b) would be entered into the Real-Time Database and those pertaining to batch transfers would be entered into the Operational History Database. Tool 6a would output a signal if there were any significant changes in the continuous flow rate.

2. On receipt of a signal from Tool 6a, the Executive would invoke a script that would seek short-term assurances by executing a number of observer/detector combinations (Tools 7a/6a) that look at specific ways in which these changes might be explained. Tool 7c would then be invoked to assess these explanations.

3. On positive change in volume of a rework tank (if monitored), confirm that it has been received from the appropriate tank; on negative change in volume of a rework tank (if monitored), confirm that it has been exported to the appropriate tank.

4. Chemical composition data for a particular tank set would be estimated by executing the appropriate simulation whenever its last buffer tank is emptied. Execution of Tool 6b would follow this.

5. On receipt of a signal from Tool 6b, the Executive would invoke a script that would seek short-term assurances by executing a number of observer/detector combinations (Tools 7a/6b) that look at specific ways in which these changes might be explained. Tool 7c would then be invoked to assess these explanations.

6. A timer would trigger the script that seeks medium term assurances, perhaps every one or two days.

7. When input accountancy tank laboratory results are obtained, a script would be invoked to update various real-time database fields.

8. A signal would be received from the Samples Database whenever relevant sample data is entered. This would trigger a script that seeks both short and medium term assurances.

4.4 The databases

It is assumed that there would be integrating software that would connect the tools together, co-ordinate input/output and interact with the inspector.

4.4.1 Nomenclature / Conventions

The plant is divided into a number of areas called segments. Each segment contains a number of plant units that are connected together. Each connection has a unique path identifier (path id) associated with it, e.g. c01t04 (i.e. cycle 1 to tank 4), together with two *transfer data streams*, Streaml(path id) describes the time history of the material entering the connection from the upstream process unit, whilst Stream2(path id) describes the time history of material entering/leaving the connection along a path other than through the normal exit into the downstream plant unit. An *input data stream* is then formed from these two transfer data streams: although the Stream2 flow rate can be positive or negative, both the Streaml and Input Stream flow rates must be positive. A diversion from the upstream unit would appear in Stream2.

A similar naming convention is used for transfers to/from hidden inventory e.g. t01h01 and for measurement errors e.g. V01 is used to denote an error in tank_1 volume. Use is also made of the 'wild card' symbol * e.g. t**h** denotes tank '*not specified*' to hidden inventory '*not specified*'.

Both transfer data streams conform to the following format:

Time (t),
Bulk flow rate (f), Bulk density (ρ_s),
Acid conc. ([H^+]), Pu conc. ([Pu]), U conc. ([U]), Other conc. ([unsp])

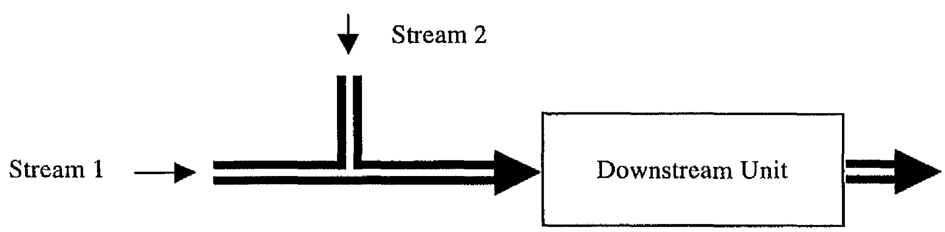


Figure 9: Data streams

If appropriate data is available that pertains to 'off-path' units, then this data would be included explicitly in the database, path identifiers would be assigned and the 'branching' would take place via Stream1 & Stream2. Otherwise these units would be classed as hidden inventories.

4.4.2 The Real-Time Database

That part of the Real-Time Database dedicated to, say, Tank 1 would look something like that shown in Table 3. Note that the streams to pertain to the connection upstream. A list of typical variables is given in Appendix 4.

	Tank 1												
Time	Volume	Density	Temp.	State 1 ...	Stream1			Stream2			and so on	
					f	$[Pu]_{in}$	....	f	$[Pu]_{in}$	.....		
0													
15													
30													
45													
60													
75													
⋮													
⋮													

Table 3: Examples of Real-Time Database fields

The Real-Time Database would include the flow histories of material imported from the input accountancy tank and exported from the product accountancy tank. It would also store the states associated with the simulations to enable them to be executed from any instance of time. As shown, the database is viewed as being flat i.e. 2-dimensional. Although space savings could be made by changing the structure of the database, this is for the implementers to decide. Here the database is kept simple. An important feature is that when a new row (i.e. time interval) is created, it is filled with values indicating 'not known' and NOT zero.

4.4.3 The Operational History Database

The Operational History Database contains that information that is not time-stamped and is hence most likely to be based on data objects. There would be three categories of object: pertaining to *declared operation*, to *events* and to *process units.*

As required by the reasoning processes, part of the declared operation might be represented by data objects that describe the statuses of certain parts of the plant. For instance, the declared-operation::cycle_1 object might look like:

declared operation::cycle_1
    status = ('loaded' 'SAF' 'empty') ; changed from empty to SAF to loaded
    times = (t_3 t_2 t_1) ; times at which status is deemed to have changed

where perhaps the current and past two statuses are stored.

An event might be represented by a class of object that has a set of sub-classes representing its possible diagnoses and symptoms:

event_id ∈ event_ids:
    event_id::event_description =
    event_id::sub-events = (sub-event_id1)

where sub-event_id1 is an instance of the Sub-event object, which has attributes:

diagnoses = (diagnosis_id1)
scores = (1)
symptoms = (symptom_id1)
diagnosis =
score =
link-with =
link-type =

where diagnosis_id1 is an instance of the sub-class, sub-diagnosis, which has attributes:

sub-diagnosis::path-type =
sub-diagnosis::path-id =
sub-diagnosis::quantity =

and symptom_id1 is an instance of a sub-class with attributes:
error-in =
path-id =
start-time =
stop-time =

Key data and the most recent history of individual plant units might be represented by the class of objects, *unit,* with attributes:

Unit_id =
Tank_type = '*buffer_tank*'
Tank_description = '*batch_transfer*'
Calibration_multiplicative_error =
event_ids =
fill_times = (start time, end time), (start time, end time) and so on.
empty_times =
volume_transfers =
mass_transfers =

But again this would be a decision for the system implementers.

It is assumed that data pertaining to samples would reside in a database elsewhere. A bridge would have to be constructed to this other database and synchronisation established so that the system is notified whenever relevant data is entered.

4.5 Simulations

The simulations are 'driven' by data streams that describe transfer time histories. Prior to the execution of the simulation, appropriate data streams would be formed by

- extracting & rationalising the relevant data from the real-time database;
- superimposing any hypothesised scenarios onto the individual data streams.

Relevant data streams for the hypothesised scenarios would be obtained from the list of events and also from any sub-events, specifically identified but not yet entered onto the list.

The computer simulation is central to a number of the tools. Essentially consisting of materials balances and hence of ordinary differential equations, the simulation would be produced using a differential equation solver. The basic structure of a differential equation solver is shown in Figure 10. The inputs would consist of a start time, t_s, a stop time, t_f, a time increment Δt, and a time vector of input variables, $\mathbf{u}(t)$. Incremental time Δt, a sub-multiple of the data collection period (T_c), would be pre-specified. The form of *Initial Condition Generator* would depend on the particular application. Two vectors are output at every time step t (= $t_s+i\Delta t$): the state vector $\mathbf{x}(t)$ (i.e. the variables that are differentiated) and $\mathbf{y}(t)$ consisting of any other variables that might be of interest.

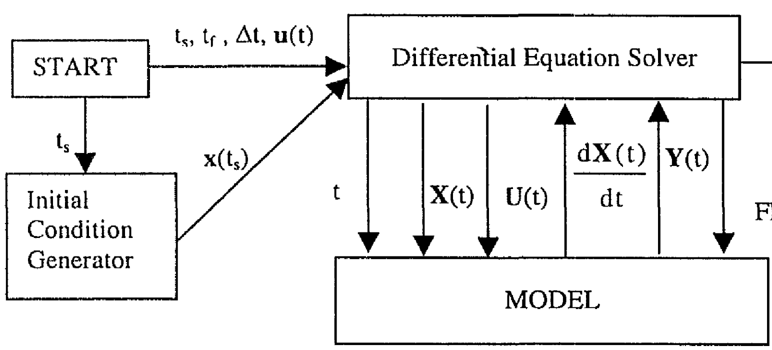


Figure 10: Computer Simulation Data Flows

The simplest form of differential equation solver, the 1st order Euler, is described here whilst, in practice a marginally more complicated 4th order Runge-Kutta might be more practicable. The algorithm is then as follows:

Start;
get $\mathbf{x}(t_s)$;
$t=t_s$;

LOOP until $t \ge t_f$:

call model with FLAG=1, $\mathbf{U}(t)=\mathbf{u}(t)$, $\mathbf{X}(t)=\mathbf{x}(t)$ to obtain $\frac{d\mathbf{X}}{dt}$;

$\mathbf{x}(t+\Delta t)=\mathbf{x}(t)+\frac{d\mathbf{X}}{dt}\Delta t$;

$t=t+\Delta t$;

call model with FLAG=3, $\mathbf{U}(t)=\mathbf{u}(t)$, $\mathbf{X}(t)=\mathbf{x}(t)$ to obtain $\mathbf{y}(t)=\mathbf{Y}(t)$.

5. SOME COMMENTS

5.1 Exploiting the Self-Validating Properties of Tank Level/Density Instrumentation

It is well-known that the level/density measurement system has a propensity to be self-validating because the two key dip-tube pressure signals are highly correlated. To reduce the complexity of the analysis system, and because it is sensible anyway, it would be preferable that faults internal to level/dip-tube instrumentation should be isolated separately. Adherence to the recently proposed standard on SEVA sensors might be beneficial here.

It might also be worth considering whether other instrumentation might be capable of outputting their own diagnostics.

5.2 Plant interruptions, data spikes and so on

Although Tool 3 can be designed to accommodate intermittent operation, Tool 2 is likely to be less flexible with the net effect that material is likely to move to and from hidden inventory, transiently. Tool 7c is designed to identify and 'remove' these events from the Operational History Database whilst Tool 6 is designed to ignore very short term transients like data spikes.

5.3 Relationship with DIE/DIV

The approach relies on process models and hence on detailed knowledge of both the topology of the plant and of key parameters. Some of this knowledge would be verified through DIE/DIV. Thus it would contribute to the specification of the DIE/DIV processes: what to verify, when to verify, priorities for verification and so on. It would also ensure that any undeclared alterations to the plant are detected once they affect plant operation.

5.4 Relationship with Near Real Time Materials Accountancy

This somewhat depends on the form of NRTMA system installed, for instance on the frequency at which balances are closed and on whether any resolution tools are included. Although there is obviously a degree of overlap, the two approaches are essentially complimentary. There are clearly similarities that are inherent because both make use of plutonium balances, although usually over different time-scales. However the system described here provides features that are explicitly different:

- it makes use of additional relationships:
  - bulk & acid balances;
  - physical relationships between variables;
- it reasons at a more microscopic level;
- it has a diagnostic capability based on the structure and function of the plant as opposed to one based on materials balances over entire MBAs.

In essence it is able to detect & diagnose certain incidents that an NRTMA system cannot, and in a more timely manner (e.g. as in Appendix 4). However the level of detail that is required to achieve this makes it cumbersome to use for longer term incidents like biases in KMP instrumentation and here NRTMA is more suitable. In addition the simplicity of the basic NRTMA approach makes it amenable to statistical analysis, the same cannot be said for the approach described here.

The main contribution that this approach would give, explicitly, to any NRTMA system is that the inspector's understanding of the contents of the plant at any instance of time increase would be enhanced.

5.5 Frequency of Decision Making

Although the system would be collecting data continuously, the analysis would be most effective if carried out over a variety of time periods:

- visibility issues would obviously be addressed virtually in real-time although follow-up activities are likely to depend on the gathering of further evidence;
- some decisions would be affected by delayed receipt of data, for instance, by samples undergoing laboratory analyses;
- the detection of small biases and the like would benefit from an additional assessment of data pertaining to a longer period of time, perhaps over the duration of an entire campaign of 12-15 days.

5.6 Software Requirements

For purposes of quality assurance, it is important that the computer programs that form the collection, analysis and presentation aspects of the system should be properly software engineered. In particular there is a need to decompose the various aspects of the system monitoring tasks in a modular fashion. The current stage merely seeks to examine the overall concepts, detailed software specification is deferred until later.

6. CONCLUSIONS

This report has outlined a methodology for confirming that a facility is operating in a way declared by the operator. It is largely unproven, based on software developments that are evolving all the time, and would require an extensive programme of work to ensure its success on a commercial facility (see below). Considerable thought has gone into ensuring that, having invested so much in its implementation, the evaluation tools can be evolved so that the system matches up to expectations. Thus there has been considerable emphasis on the kernel, composed of data stored in two databases, and hence on the quality of the data collection system. It is important that appropriate data is collected; although the analytical tools can evolve, the data cannot i.e. one cannot go back and collect the same data again. Similarly it is important that both the structure of the databases and the specification of their numerous data fields are carefully thought about; it is difficult to re-build a database once it has started to be used.

Results from various case studies have demonstrated that the application of such a concept would provide assurances that the plant is operating as declared. These case studies have focused on demonstrating how the system would work rather than on measuring the system's ability to detect and isolate a particular incident. This is partly because quantitative results would be misleading unless they were based on data that look like real data, which we do not have, and partly because of a lack of resources.

6.1 Recommended Route To Implementation

The system outlined here has been developed with implementation in mind. Since this type of system has never been implemented before, it is important that it is structured in a way that facilitates testing, feedback and further development (i.e. evolution). Certain aspects are easier to change than others. Going from the most inflexible to the least inflexible, these are listed below:

- the collection of measurement data;
- the representation of collected data in databases;
- the representation of other data in data bases;
- the specification of data to be stored in the databases;
- outlines of the types of method (procedure) that will interact with the database;
- user interface development tools;
- the methods;
- user interfaces.

This evolutionary process would involve the testing of alternative versions of the tools.

There are three strands to the implementation: data collection and pre-processing, database and user interface construction, and the tools described here. Although all three do not need to start at the same time some degree of concurrency is essential. The kind of overlap that is anticipated is shown in Table 4 below.

Period	1	2	3	4	5
Plant commissioning phase	Water Runs	Uranium Runs		Low Burn-up Runs	
Data collection and pre-processing					
Database/user interface construction					
The tools					

Table 4: Implementation

6.1.1 Data Collection and Pre-processing

The form of the data, what it looks like and how it is stored, can have a considerable effect on the design of the tools. In particular data compression is a major issue; information lost as a result of incorrect pre-processing can never be retrieved. It is therefore important that data storage receives proper consideration. As a first step the profiles of individual instruments should be recorded and analysed, with the end use in mind, as early as possible. Here some degree of testing of the tools will be necessary. The implementers need not wait for the plant to become operational to obtain these profiles, data obtained from test facilities and during the various stages of commissioning could be analysed to enable the system to be developed in a sensible manner.

Work on providing the level/density instruments with a self-validating capability should be carried out in parallel. Considerable expertise in developing software for these instruments already exists at Ispra who also have the facilities for testing the product.

6.1.2 Database/User Interface Construction

There are a number of different types of database on the market. In selecting an appropriate database it is important to appreciate that the requirements here are somewhat different to the requirements for which most of these products were developed. In particular with the so-called 'real-time databases', which have data compression, display and archiving schemes. The emphasis is often on data presentation rather than on providing a bridge to an automatic analysis capability.

6.1.3 The Tools

Various versions of the individual tools are proposed to accommodate the fact that it is not possible to design the various analysis tools without testing using realistic data. Real data is needed to both develop and test the approach. The analytical procedures are being developed on the basis of simulations. Since there is a considerable difference between simulated data and reality, it is important that the tools should be tested and refined using real-data. A programme of activities should be organised to provide the appropriate data. Initially this might come from the THAME facility at Ispra or from water tests at RRP. Later on, the opportunity should be taken to collect data at each stage of commissioning.

7. REFERENCES

1. J.P. Shipley. DECANAL User's Manual, LA-9043-M, 1982.
2. M.H. Ehinger. Process Monitoring in International Safeguards For Reprocessing Plants – A Demonstration, ORNL/TM-10912, 1989.
3. M.H. Ehinger, N.R. Zack, E.A. Hakkila & F. Franssen. Use of Process Monitoring for Verifying Facility Design for Large-Scale Reprocessing Plants, LA-12149-MS / ORNL/TM 11856, 1991.
4. G. Andrew, Safeguards – Changes and Challenges, *JNMM* 26(2), pp. 12-16, 1998.
5. JNFL Project. User Requirements and Functional Specifications for the Solution Monitoring System at the Rokkasho Reprocessing Plant. IAEA, July 1999.
6. E.C. Miller & J. Howell. Tank Measurement Data Compression For Solution Monitoring *JNMM* 27(3), pp 25-32, 1999.
7. J.V. Candy and R.B. Rozsa. Safeguards Design for a Plutonium Concentrator- An Applied Estimation Approach, *Automatica*, 16, pp 615-622, 1980.
8. F.Franssen, Private communication
9. R.Abedin-Zadeh, Private communication.
10. T. Burr & L. Wangen. Enhanced Safeguards Via Solution Monitoring, LA-13186-MS, 1996.
11. J.P. Dekens, J. Goerten, Y. Lahogue & H.G. Wagner, An Integrated Safeguards System for Large Reprocessing Plants, *ANS 5th Int. Conf. On Facility Operations-Safeguards Interface*, Wyomong, 1995.
12. D. Luenberger. An introduction to Observers, *IEEE Transactions on Automatic Control*, ac-16(6), December 1971.
13. S.J. Scothern & J. Howell, A Physical-Model-Based Diagnostic Aid for Safeguarding Nuclear Material In a Liquor Storage Facility, *JNMM* 25(4), pp 20-29, 1997.
14. J. Howell & E. Miller, Tailoring The Glasgow University Diagnostic Aid For The Product Storage Facility At TRP, UK Safeguards Support for the IAEA, SRDP-R280, 2001.
15. K. Whitehouse. FissTrack. Presented to the IAEA training course on Reprocessing Safeguards Instrumentation, BNFL Sellafield, 1999.
16. M. Ward. Private Communication, 1999.
17. P.M. Frank. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy – A survey and some new results, *Automatica*, 26, pp 459-474, 1987.
18. R. Isermann. Supervision, fault-detection and fault-diagnosis methods – An introduction, *Control Engineering Practice*, 5, pp 639-652, 1997.
19. J. Howell. Model-based fault detection in information poor plants. *Automatica*, **30(6)**, pp 929-943, 1994.
20. T.L. Burr, L.E. Wangen & M.F. Mullen, Authentication of Reprocessing Plant Safeguards Data through Correlation Analysis. Los Alamos National Laboratory, LA-12923-MS, 1995.
21. J. Howell & S.J. Scothern, Assessing Solution Monitoring System Performance Using Simulated Data, UK Safeguards R&D Report, SRDP R261, 1999.
22. C. Chatfield, *Statistics for Technology: A Course in Applied Statistics*, third edition (revised). Chapman and Hall, London, 1996.
23. D.C. Montgomery, *Introduction to Statistical Quality Control*, third edition. Wiley, Chichester, 1996.
24. P.J.M. Van Laarhoven. *Theoretical and computational aspects of simulated annealing,* CWI Tract 51, Centrum voor Wiskunde en Informatica, Amsterdam, Netherlands, 1988.
25. L.P.P.P. Van Ginneken & R.H.J.M. Otten. *The Annealing Algorithm*, Kluwer Academic Publishers Group, Dordrecht, Netherlands, 1989.

26. G.S. Fishman. *Monte Carlo, Concepts, Algorithms, and Applications*, Springer Series in Operations Research, Springer-Verlag, New York, United States of America, 1996.
27. I.O. Bohachevsky, M.E. Johnson & M.L. Stein. Generalized Simulated Annealing for function Optimization, *Technometrics*, 28, No.3, August 1986.
28. S.G. Louie & D. Vanderbilt. A Monte Carlo Simulated Annealing Approach to Optimization over Continuous Variables, *Journal of Computational Physics*, 56, 1984.
29. G.T. Barkema & M.E.J. Newman, *Monte Carlo Methods in Statistical Physics*, Clarendon Press, Oxford, United Kingdom, 1999.
30. J. Howell & E.C. Miller. Estimation of Gross Systematic Multiplicative Biases in Reprocessing Plant In-Process Tanks, UK Safeguards Support for the IAEA, SRDP-R281, 2001.

APPENDICES

APPENDIX 1: ACCESSING REDUNDANT INFORMATION

Redundant information pertaining to plant operation can take two forms: derived from *physical redundancy* and *analytical redundancy*. Physical redundancy derives from there existing multiple sensors, which is uncommon in reprocessing plants. Analytical redundancy exploits the inherent static and dynamic relationships among the measured variables. In other words one makes use of a *mathematical model*. Three typical simple models are given as examples below.

1. Consider the masses of liquor in two connected tanks, tanks i & i+1, when a batch is transferred between them with no 'carry-over':

Mass of batch output from exporting tank	=	*Mass of batch imported into receiving tank*

$$Mass_i(at\_end\_of\_transfer) = Mass_i(at\_beginning\_of\_transfer) - Mass\_transferred$$
$$Mass_{i+1}(at\_end\_of\_transfer) = Mass_{i+1}(at\_beginning\_of\_transfer) + Mass\_transferred$$

Thus a measured change in level in a particular tank can be corroborated by looking for corresponding changes in neighbouring tanks.

2. The relationship between level and the flow rates in and out of a tank that is exporting/importing continuously is governed by

$$Mass(t) = Mass(0) + \int_0^t (flow\_rate\_in(t) - flow\_rate\_out(t))\, dt$$

Thus a change in the level in a tank can be corroborated if its net flow rate is measured.

3. Consider the masses of liquor in two tanks, tanks i & i+1, where there are now continuous transfers both in & out of either of them; assume once more that there is a constant hold-up in the inter-connecting pipe, the following mass balances can be formed:

$$Mass_i(t) = Mass_i(0) + \int_0^t flow\_rate\_in_i(t) - flow\_rate\_in_{i+1}(t)\, dt$$
$$Mass_{i+1}(t) = Mass_{i+1}(0) + \int_0^t flow\_rate\_in_{i+1}(t) - flow\_rate\_out_{i+1}(t)\, dt$$

Once again the levels in the two tanks will be related, this time through the common variable $flow\_rate\_in_{i+1}$.

In practice, accurate measurements of individual flow rates might not be available and a tank might be importing and exporting at the same time making simple level comparison difficult. Whilst keeping to the same principles, it is more sensible to reason about the models more explicitly. Amongst the numerous model-based detection and diagnostic techniques that have been developed to reason about models (see, for instance, references 17-19), a number have been developed with the specific needs of monitoring for nuclear safeguards in mind (see, for instance, references 20 & 21).

APPENDIX 2: ASSURANCES IN OTHER AREAS

A2.1 Accountancy tanks

A few years ago now, Euratom announced that they were developing a system that would obtain additional assurances about the operation of the input and output accountancy tanks[11]. The approach was essentially one of pattern matching where the actual volume history was compared against templates. Such an approach would integrate well with the overall system proposed here. Unfortunately the authors are unaware of the current situation so cannot comment further.

A2.2 Plutonium Nitrate Product Storage Area

This area has not been considered because it has been examined, extensively, elsewhere [13]. Unfortunately the objectives behind that work were somewhat different to those that were specified for the system outlined in this report. A somewhat simpler approach (see Recommendation 4 in [14]) would be more appropriate, but this would require further development.

A2.3 Finishing Plant

The tanks containing solutions at the start of this plant can be treated in a similar way to those in the process plant. The instrumentation available in other parts of the plant can largely consist of neutron detectors, weighing devices, environmental temperature sensors and environmental relative humidity sensors, all of which would be of interest to a data evaluation system. BNFL report considerable success with their neutron detector-based PIMS/FissTrack based systems for plant monitoring [15], and there is a clear synergy with the approach described here. Although its use as a safeguards tool appears to be contentious [16] and further development work would be required so that it could be integrated with the evaluation tools, it is clear that additional assurances might be got. One possible (untested) approach is broadly outlined below:

Tool 5: driven by Operator Declarations – Category B, by the feed flow rate (Tool 3) and by weighings, a simple plant simulation composed of mass balances would be executed to predict the instantaneous mass distribution of material throughout the plant.

Tool 6: this mass distribution would be compared with that estimated by the on-line PIMS/FissTrack System and any disagreements would be signalled.

Tool 7: taking into account environmental temperature and relative humidity, a model-based diagnostic system would hypothesise reasons for these disagreements.

APPENDIX 3: EXECUTIVE DECISIONS

The Tools are co-ordinated by the Executive, the precise form of which would depend on the software platform installed.

A3.1 Decision Boxes

Decision Box 1

(In practice this link is likely to be included in Tool 3a. It is shown as a separate box to enable the reader to see a key (asyncronous) feature of the system).

When Tool 3a-1 outputs 'tank_id, $t_{s,i}$, $t_{f,i}$ ',
Tool 3b-1(tank_id, $t_{start} = t_{s,i}$,$t_{finish} = t_{f,i}$)

Decision Box 2

When the buffer tank upstream of the cycle feeding tank is emptied,
Tool5(Segment_id = Tank_set upstream,
$t_{start} = t_{s,i}$, pertaining to this tank
$t_{finish} = t_{f,i}$ pertaining to this tank)
Tool 6b

When the buffer tank with the longest nominal cycle time completes a cycle,
wait until all Tool 1a procedures that pertain to the tank_set upstream output 'no transfer', Tool5(Segment_id = 'Tank_set upstream',
$t_{start} = t_{s,i}$, pertaining to this tank
$t_{finish} = t_{f,i}$ pertaining to this tank)
Tool6b

Decision Box 3

When input accountancy tank laboratory results are obtained,
update the following real-time database fields:
Stream 1 of the buffer tank downstream of the accountancy tank:
f, ρ_s, [Pu], [U], [H^+], [unsp] .

A3.2 Short-term assurances

When (Tool 6a-1 outputs a non-empty set sub-event_ids) &
(sub-event_ids pertain to a receiving tank),

$\forall$ sub-event_id $\in$ sub-event_ids,

t_{start} = sub-event_id::symptoms::start_time
t_{finish} = sub-event_id::symptoms::stop_time
Segment_id = unit-upstream(sub-event_id::symptoms::path_id)

if $\exists$ symptom_id $\in$ sub-event_ids::symptoms : symptom_id::error-in = 'Flow'

[I_{hinv}] = Tool 7a(Segment_id, 'Hidden_inventory', t_{start}, t_{finish})
Tool 6a-2(I_{hinv}, t_{start}, t_{finish}, sub-event_ids)
[Pu, H^+] = Tool 7a(Segment_id, 'Pu/H^+_out', t_{start}, t_{finish})
Tool 6a-3((Pu, H^+), t_{start}, t_{finish}, sub-event_ids)
[Pu, unsp] = Tool 7a(Segment_id, 'Pu/unsp_out', t_{start}, t_{finish})
Tool 6a-4((Pu, unsp), t_{start}, t_{finish}, sub-event_ids)

else

[Pu] = Tool 7a(Segment_id, 'Interpret Pu', t_{start}, t_{finish})
Tool 6a-5(Pu, t_{start}, t_{finish}, sub-event_ids)
[H^+] = Tool 7a(Segment_id, 'Interpret H^+', t_{start}, t_{finish})
Tool 6a-5(H^+, t_{start}, t_{finish}, sub-event_ids)
[unsp] = Tool 7a(Segment_id, 'Interpret unsp', t_{start}, t_{finish})
Tool 6a-5(unsp, t_{start}, t_{finish}, sub-event_ids)

end

Tool 7c(sub-event_ids)

When (Tool 6b-1 outputs a non-empty set sub-event_ids) or
((Tool 6a-1 outputs a non-empty set sub-event_ids) &
(sub-event_ids pertain to a feeding tank)),

$\forall$ sub-event_id $\in$ sub-event_ids,

Segment_id = The id of the Tank set that contains the tank
t_{start} = sub-event_id::symptom_id1::start-time
t_{finish} = sub-event_id::symptom_id1::stop-time
unit_downstream(sub-event_id::symptoms::error-in) $\rightarrow$ unitd
$t_{fin} = 0$,
found = .nil.

Tool 7a(Segment_id, 'Transfer_To_Hinv', t_{start}, t_{finish}) $\rightarrow$ [t_s, t_f, f]
merge(unitd Stream 1, Stream 2 & Input Stream with f) $\rightarrow$
revised_stream $\in$ revised_streams
path_id pertaining to unitd $\rightarrow$ revised_id $\in$ revised_ids
Tool5(Segment_id, t_s, t_f, revised_ids, revised_streams) $\rightarrow$
[measurement_predictions]
Tool 6b-2(Segment_id, t_s, t_f, measurement_predictions, sub-event_id, f) $\rightarrow$
success
if success $\rightarrow$ $t_f = \min(t_f, t_{fin})$, found = .t.

% Repeat for observer_type = 'Transfer_In'
Tool 7a $\rightarrow$ [t_s, t_f, f]
merge $\rightarrow$ revised_stream $\in$ revised_streams
path_id pertaining to buffer tank $\rightarrow$ revised_id $\in$ revised_ids
Tool5 $\rightarrow$ [measurement_predictions]
Tool 6b-2 $\rightarrow$ success
if success $\rightarrow$ $t_f = \min(t_f, t_{fin})$, found = .t.

% Repeat for 'Addition_of_Acid'
Tool 7a $\rightarrow$ [t_s, t_f, f, c]
merge $\rightarrow$ revised_stream $\in$ revised_streams
path_id pertaining to unitd $\rightarrow$ revised_id $\in$ revised_ids
Tool5 $\rightarrow$ [measurement_predictions]
Tool 6b-2 $\rightarrow$ success
if success $\rightarrow$ $t_f = \min(t_f, t_{fin})$, found = .t.

% Repeat for 'Addition_of_Pu'
Tool 7a $\rightarrow$ [t_s, t_f, f, c]
merge $\rightarrow$ revised_stream $\in$ revised_streams
path_id pertaining to unitd $\rightarrow$ revised_id $\in$ revised_ids
Tool5 $\rightarrow$ [measurement_predictions]
Tool 6b-2 $\rightarrow$ success
if success $\rightarrow$ $t_f = \min(t_f, t_{fin})$, found = .t.

```
%       If all observers fail, mark as a measurement error
        if not( found)
                sub-event_id::diagnoses = (diagnosis_id1)
                sub-event_id::symptom_id1::start-time =
                sub-event_id::symptom_id1::stop-time =
        end


%       Continue to search
        if call was from procedure 6b-1 & t_f < t_finish
                Tool5(Segment_id, t_f, t_finish, { }, { }) → [measurement_predictions]
                Tool 6b-1(Segment_id, t_f, t_finish, measurement_predictions) →
                                                                    sub-event_id
        end

  end  {sub-event_id loop}

  Tool 7c(sub-event_ids)
```

A3.3 Medium-term assurances

When timer = specified_time,

Period_id = period_id + 1

[problem_unit error redistributions] = Tool7b(Period_id, t_{start}, t_{finish})

```
if problem_unit = null
        redistribute(t_start, t_finish, redistributions)
        Tool 7b-4
else
        create a sub-event, sub-event_id ∈ medium-term-sub-event_ids:
                diagnoses = (diagnosis_id1)
                symptoms = (symptom_id1)
                diagnosis_id1 has a sub-diagnosis sd_id1 which is a sub-class:
                        path-type =     'flow'
                        path-id   =     if error<0 then
                                                path: problem_unit to hidden inv
                                        else
                                                the same path in reverse
                symptom_id is a sub-class:
                        error-in  = 'redistribution'
                        path-id   = problem_unit
                        stop-time = t_finish
        Tool 7c(medium-term-sub-event_ids)
end
```

Increase specified_time by N days.

NB If problem_unit contains more than one item (i.e. points to a more general area), use a special name.

A3.4 Analysing sample data

When data pertaining to a relevant sample is written to the sample database,

get time = t_{sample} , Tank_id, $[Pu]_s$, $[U]_s$, $[H^+]_s$, $[unsp]_s$, ρ_s , T ;
update density correlation ;

obtain from real-time database for time = t_{sample}: M_{bulk}, M_{Pu}, M_U, M_{acid}, M_{unsp} ;

substitution-with-acid? = .false. ;
if $\left[M_{Pu} - M_{bulk}[Pu]_s\right] > 1kg$ then substitution-with-acid? = .true. ;

obtain from Tank_id::fill-times the time, t_s, at which the tank was previously started to be filled;

overwrite all the M_{Pu}, M_U, M_{acid}, M_{unsp} entries in the real-time database, for Tank_id, for the time period t: $t_s < t \le t_{sample}$:

$M_{Pu} = [Pu]_s M_{bulk}$ and so on ;

if substitution-with-acid?

then

Segment_id = The id of the Tank set that contains the tank
Tool 7a(Segment_id, 'Substitution_with_acid', t_s, t_{sample}) → [t_s, t_f, f]

create a sub-event, sub-event_id ∈ medium-term-sub-event_ids:
diagnoses = (diagnosis_id1)
symptoms = (symptom_id1)
diagnosis_id1 has a sub-diagnosis sd_id1 which is a sub-class:
path-type = measurement_error
path-id = s**
quantity = see test above
symptom_id is a sub-class:
error-in = sample
path-id = Derived on the basis of Tank_id
stop-time = t_{sample}

merge(Stream 1, Stream 2 & Input Stream with f) →
revised_stream ∈ revised_streams
path_id pertaining to input to Tank_id → revised_id ∈ revised_ids
Tool5(Segment_id, t_s, t_f, revised_ids, revised_streams) →
[measurement_preds]
Tool 6b-2(Segment_id, t_s, t_f, measurement_preds, sub-event_id, f)
→ success

end

Examine the medium term up to time t_{sample}.
Start time, $t_{start} = t_{sample} - 3*24*60$ (i.e. 3 days)

```
[problem_unit error redistributions] = Tool7b-1(t_start, t_sample)

if problem_unit = null
        redistribute(t_start, t_finish, redistributions)
else
        create a sub-event, sub-event_id ∈ medium-term-sub-event_ids:
                diagnoses = (diagnosis_id1)
                symptoms = (symptom_id1)
                diagnosis_id1 has a sub-diagnosis sd_id1 which is a sub-class:
                        path-type =    'flow'
                        path-id   =    if error<0 then
                                               path: problem_unit to hidden inv
                                       else
                                               the same path in reverse
                symptom_id is a sub-class:
                        error-in  = 'redistribution'
                        path-id   = problem_unit
                        stop-time = t_finish
        Tool 7c(medium-term-sub-event_ids)
end
```

A3.5 Local procedures

procedure redistribute(t_{start}, t_{finish}, redistributions)

$\forall$ redistribution $\in$ redistributions,
  if redistribution $\neq 0$
    from the appropriate database, obtain the total Pu transferred out of this unit
    over the time period : $t_{start} \leq t < t_{finish} \rightarrow M_{Pu}$
    calculate % correction: redistribution/M_{Pu}
    apply this correction to the associated volume transfers in the databases

APPENDIX 4: SOME DATABASE FIELDS

Location	Type	Variable	Value	SD	Health
Receiving/feeding tank	**Measurements**	Volume	✔	✔	✔
		Density	✔	✔	✔
		Temperature	✔	✔	✔
	Stream1	f, ρ_s, [Pu], $[U]^*$, $[H^+]$, [unsp]	✔		
	Stream2	f, ρ_s, [Pu], $[U]^*$, $[H^+]$, [unsp]	✔		
	Input Stream†	f, ρ_s, [X], [Pu], $[U]^*$, $[H^+]$, [unsp]	✔		
	States	M_{bulk}, M_{acid}, $M_U{}^*$, M_{Pu}, M_{unsp}	✔		
Buffer tank	**Measurements**	Volume	✔	✔	✔
		Density	✔	✔	✔
		Temperature	✔	✔	✔
	Stream1	f, ρ_s, [Pu], $[U]^*$, $[H^+]$, [unsp]	✔		
	Stream2	f, ρ_s, [Pu], $[U]^*$, $[H^+]$, [unsp]	✔		
	Input stream1*	f, ρ_s, [X], [Pu], $[U]^*$, $[H^+]$, [unsp]	✔		
	States	M_{bulk}, M_{acid}, $M_U{}^*$, M_{Pu}, M_{unsp}	✔		

† Inherently corrected to tank temperature
* Inherently corrected to temperature of tank upstream
* Tank set 1 only (plus receiving/buffer tanks in separated uranium line)

Table 5: Key database fields

Location	Type	Variable	Value	SD	Health
Input accountancy tank	**Measurements**	Volume	✔	✔	✔
		Density	✔	✔	✔
		Temperature	✔	✔	✔
Product accountancy tank	**Measurements**	Volume	✔	✔	✔
		Density	✔	✔	✔
		Temperature	✔	✔	✔
	States	Vbulk, M_{acid} ,M_U^* , M_{Pu} ,M_{unsp}	✔		

Table 5 (continued) : Key database fields

Object	Sub-class	Arrays attached to sub-class
Input accountancy tank	Transfer_id	bulk volume transferred out
		Pu concentration
		U concentration
		acid concentration
		time of transfer
Product accountancy tank	Transfer_id	bulk volume transferred out
		Pu concentration
		acid concentration
		time of transfer

Table 5 (continued): Key database fields

Location	Type	Variable	Value	SD	Health
Solvent-extraction cycle	**Measurements**	Active feed in	✔		
		Neutron detectors	T/F		
		XRF detectors	T/F		
		Inactive feed flowmeters	✔*		
	Stream1	f	✔		
	Stream2	f, [Pu], [U], [H^+], [unsp]	✔		
	Input stream	f, ρ_s	✔		
	Other	Advised nominal inventory	✔		
		Inactive feed flowmeters	I/S/D		
		Electrical conductivity (molarity)	I/S/D		
	States	Nominal inventory	✔		
		Hidden Inventory	✔		

* might be of use to Tool 2
I/S/D – increasing / decreasing / steady (Tool 8)

Table 5 (continued): Key database fields

Location	Type	Variable	Value	SD	Health
Concentrator	**Measurements**	Rate of heat input	✔*		
		Plutonium concentration at outlet	✔		
	Stream1	f	✔		
	Stream2	f, [Pu], [U], [H^+], [unsp]	✔		
	Input stream	f, ρ_s	✔		
	Other	Advised nominal inventory	✔		
		Rate of heat input	I/S/D		
	States	Nominal inventory	✔		
		Hidden inventory	✔		

* might be of use to Tool 2

Table 5 (continued): Key database fields

APPENDIX 5: THE TOOLBOXES

Toolbox 1:	Estimation of transfers into/out of a buffer tank.
Toolbox 2:	Generates nominal plutonium inventories for the solvent-extraction cycles and for the concentrator
Toolbox 3:	Estimation of bulk flow rate – receiving/feeding tanks only.
Toolbox 4:	Estimation of the X and acid components in the continuous stream into a receiving tank.
Toolbox 5:	Plant simulations.
Toolbox 6:	Disagreement detection.
Toolbox 7:	Model-based reasoning.
Toolbox 8:	Confirmation of Operational Unit Statuses

Table 1 (repeated)

A5.1 Recursive least squares

A Recursive Least Squares (RLS) algorithm [22] is used to estimate the gradient, m, and constant, c of a straight line [i.e. m,c : $y_i = mz_i + c$] as follows:

$$x_{n+1} = x_n + P_{n+1}\, h_{n+1}\, (\, y_{n+1} - h^T_{n+1}\, x_n\,)$$

where :

$$y = h^T x + n$$
$$h = [z\ 1]^T$$
$$x = [m\ c]^T$$
$$n \text{ - noise}$$
$$P_{n+1} = P_n - P_n\, h_{n+1}\, (\, 1 + h^T_{n+1}\, P_n\, h_{n+1}\,)^{-1}\, P_n\, h^T_{n+1}$$

To apply this, $P_o = \infty$ * 2x2 identity matrix and x_0 has m = 0 and c = the first data point. It is necessary to reset the RLS calculation every time a point of change is detected. This is easily achieved by simply resetting P_{n-1} to its initial value and setting X to have m = 0 and c = value of process mean at this point.

A5.2 The standard Cusum test

From reference 23, expressions for the Upper Cusum (C^+) and the Lower Cusum (C^-), when applied to a normalised data point $\frac{x_i}{\sigma_i}$, are as follows:

$$C_i^+ = max\left[0, \frac{x_i}{\sigma_i} - (\mu_o + K) + C_{i-1}^+\right]$$

$$C_i^- = max\left[0,\,(\mu_o - K) - \frac{x_i}{\sigma_i} + C_{i-1}^-\right]$$

where μ_o is the target process mean, σ_i is the standard deviation and K is a constant.

If either C_i^+ or C_i^- exceed appropriate tolerances, then the process mean is deemed to be changing. This procedure is often represented by a V-mask (Figure 11) which is applied at every point of the data; if one of the arms of the V-mask intercepts a data point then a change in the mean is deemed to have taken place. Parameters H and K relate to the vertical height and the angle of the arms of the V-mask respectively. If K = 0 the arms are at right angles. Parameter H: 1 < H < 5.

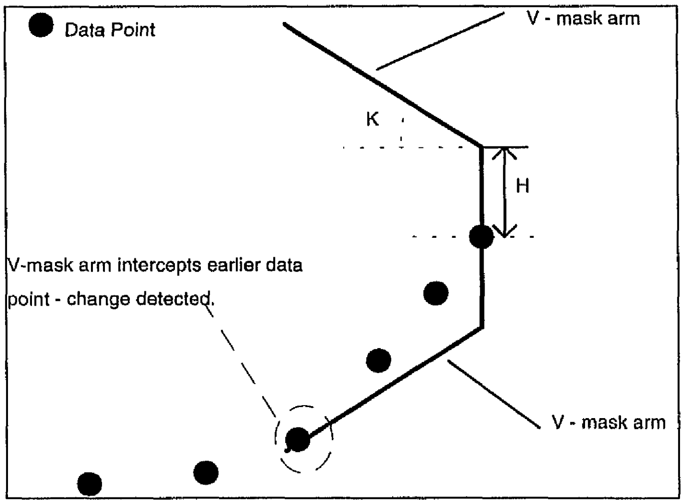


Figure 11: Illustration of V-mask

Adapting The Cusum Test To Detect A Change In Gradient

This procedure can be modified to detect whether or not the data has deviated from a straight line trajectory and, approximately, at what time this has occurred. In this case μ_o is replaced by $mz_i + c$ where $z_i = t_i - t_1$ and t_i pertains to the time at which the i^{th} point was recorded.

TOOLBOX 1: ESTIMATION OF VOLUME TRANSFERS FOR TANKS THAT BOTH IMPORT AND EXPORT NON-CONTINUOUSLY

T1.1 Tool 1a: Normal tanks.

Inputs:

Time	t
Volume	V
Density	ρ

All pertain to the exporting buffer tank.

Outputs:

To Real-Time Database:

$f_{out} \rightarrow$ f(downstream unit, stream1)
$f_{in} \rightarrow$ f(tank, input stream)

To Operational History Database:

if $f_{out} > 0$ then $t_{s,i}$, $t_{f,i}$, Volume_transferred $\rightarrow$ tank object
if $f_{in} > 0$ then $t_{s,i}$, $t_{f,i}$, Volume_transferred $\rightarrow$ upstream tank object

This tool would be executing all the time. The procedure, which is outlined below, contains a number of refinements. It is recommended that the first version omits Stages 4 (i.e. assume a constant σ), 7 & 8. Subsequent versions (with Stage 4 or 7 & 8 or 4, 7 & 8) can then be tried to see whether the added complexity brings extra benefit.

The procedure makes use of the following:

a ring buffer **p** (of size 2400 ≡ 10 hours ?)
a shuffle buffer **mc** (of size 5 ?)
variable *tranfers* initialised with the first data point collected
variable *offset* initialised at 0
parameters $\eta_{suspend}$ (typically 20), F_U, F_L , λ (typically 0.1), tol, N_s
(F_U and F_L are typically integers < 6 and need not be equal)

1. Receive data point from database and place in ring buffer, **p**:(t, V, ρ) → p ∈ **p**
2. Apply RLS algorithm to V → (m, c)
3. Pop (m, c) into the bottom of the shuffle buffer, **mc** and collect (m', c') which pops out of the top as a consequence of doing this.
4. Update the <u>estimate</u> of the mean square error of the data:

$$\lambda(V - m' t - c')^2 + (1-\lambda)\sigma^2 \rightarrow \sigma^2$$

5. Calculate the upper control limit (UCL) and lower control limit (LCL):

$$\tilde{V} = m' t + c'$$
$$UCL = \tilde{V} + \sigma F_U$$
$$LCL = \tilde{V} - \sigma F_L$$

6. Test for change in direction

If V > UCL

% Increase detected
Suspend both UCL and LCL tests for $\eta_{suspend}$ data points.
Reset RLS: m → 0 ; c → V ; $P_n \rightarrow P_0$;
(p m' c') → **points**

end

If V < LCL

% Decrease detected
Suspend both UCL and LCL tests for $\eta_{suspend}$ data points.
Reset RLS: m → 0 ; c → V ; $P_n \rightarrow P_0$;
(p m' c') → **points**

end

7. When (p m' c') written to **points**, generate estimates for intercept on the basis of the last two points (i.e. do nothing if only one element in points:

size = size-of(**points**)

If size > 1
using the last two points stored i.e. $(t_i\ V_i\ \rho_i\ m_i\ c_i)$, $(t_{i-1}\ V_{i-1}\ \rho_{i-1}\ m_{i-1}\ c_{i-1})$:

$$t_{\text{int}} = \frac{c_i - c_{i-1}}{m_i - m_{i-1}}$$
$$V_{\text{int}} = m_i t_{\text{int}} + c_i$$

end

8. Obtain $(V_{int} - V)^2$ for each of the $\frac{N_s}{2}$ data points $\in$ **p** whose times are just before $t = t_{int}$ and each of the $\frac{N_s}{2}$ data points $\in$ **p** just after $t = t_{int}$. Hence find the data point with the smallest $(V_{int} - V)^2$ and place the previous data point from **p** (i.e. one with position i_{-1}) into the array **transfers** at position (i_{-1}-offset). Pass position i to Stage(9).

9. When position i received, construct a transfer pair from points stored in **transfers:**

```
ii = 0
for jj = 1:[size(transfers)-1]
        if sign(m_jj ∈ transfers) ≠ sign(m_(jj+1) ∈ transfers)
                ii = ii +1
                pair(ii) = transfers(jj)  ; i.e. pass the point located at jj
        end
        if ii = 2
                % have a transfer pair
                last_point = last entry in transfers
                ii = 0
                clear transfers
                transfers(1) = last_point
                offset = i - 2

                % pass pair to stage 10
                pair → stage(10)
        end
        if ii < 2 and jj = size(transfers)-1
                % no pair available in transfers
                ii = 0;
        end
end
```

10. When **pair** received, calculate volumes transferred etc.:

Volume_transferred = $\text{Volume}_2 - \text{Volume}_1$;

Mass_transferred = $\text{Density}_2 * \text{Volume}_2 - \text{Density}_1 * \text{Volume}_1$;

flow rate = $\dfrac{Volume_2 - Volume_1}{\Delta T}$ where $\Delta T = \Delta t * \text{int}\left(\dfrac{t_{f,i} - t_{s,i}}{\Delta t}\right)$ and Δt is the integration time step of the computer simulation;

{ $f_{in} = f_{out} = 0$,
  for the period of time t: $t^*_{s,i} \le t < t^*_{s,i} + \Delta T$ where $t^*_{s,i}$ is the last time point in the database before $t_{s,i}$,
    if flow rate > 0 then
      f_{in} = flow rate
    else
      f_{out} = flow rate
end }.

T1.2 Tool 1b: The input accountancy and product accountancy tanks

Little can be written about this particular tool until the precise mode of operation of each tank is known. Conventional accountancy data would be used when it became available:

When accountancy data is written to the conventional inspector's database,
obtain times transfer started and stopped: t_1 , t_2
obtain values at start of transfer: $Volume_1$, $Density_1$
obtain values at end of transfer: $Volume_2$, $Density_2$

Then as per Step 10. above

Obtain the chemical composition of the solution transferred,. [Pu] etc.

Then
$\forall t$: $t_1 < t \le t_2$, [Pu] $\rightarrow$ [Pu] (downstream unit, stream 1) etc.

Prior to receiving this data, estimates would be obtained, probably as per Tool 1a. Chemical compositions would be assumed to be the same as for the previous batch.

TOOLBOX 2: GENERATES NOMINAL PLUTONIUM INVENTORIES FOR THE SOLVENT-EXTRACTION CYCLES AND FOR THE CONCENTRATOR

Inputs:

Outputs:
- To Real-Time Database: nominal inventory
- To Operational History Database:

The authors have been told that Tool 2 is to be developed elsewhere. Presumably the operator will make available correlations for estimating the plutonium inventories in each of the cycles and the concentrator. These correlations might depend on a flow sheet as declared as part of the Declared Operation or might depend on measured variables, which will have to be placed in the real time database. Tool 2 will evaluate the correlations and output to the Real-Time Database.

TOOLBOX 3: ESTIMATION OF BULK FLOW RATE - RECEIVING/FEEDING TANKS ONLY.

Inputs: Volume
Outputs:

To Real-Time Database:

$f_{out} \rightarrow f$ (downstream unit, stream 1)
$f_{in} \rightarrow f$ (receiving/feeding tank, input stream)

To Operational History Database:

α_i , $t_{s,i}$, $t_{f,i}$ $\rightarrow$ receiving/feeding tank object

T3.1 Tool 3a Procedure.

Tool 3a continuously records the incoming data in a buffer; emptying the data stored within whenever the start of a new cycle is detected. It is recommended that the first version omits Stages 4, 5 (i.e. assume a constant σ), and 6. Stage 6 can be replaced with Stages 5 and 6 from Tool 1 (i.e. use the combined Shewhart/RLS detector). Subsequent versions can then be tried to see whether the added complexity can be justified in terms of the improved performance.

The procedure makes use of the following:

a shuffle buffer **mc** (of size 5 ?)
a shuffle buffer **std** (of size 5 ?)
a dynamic buffer **cycle**
record = 1 @ t = 0, recording flag for cycles
switch = flow 'switching time' of cycle
type = 1 for receiving tank
type = -1 for feeding tank
parameters: $\eta_{suspend}$ (typically 20)

1. Receive data point from database, (t, V, ρ)
2. Apply RLS algorithm to $V \rightarrow (m,c)$
3. Pop (m,c) into the bottom of the shuffle buffer, **mc** and collect (m',c') which pops out of the top as a consequence of doing this.
4. Calculate standard deviation of $V \rightarrow \sigma$
5. Pop σ into the bottom of the shuffle buffer, **std** and collect σ' which pops out of the top as a consequence of doing this.

6. Apply CUSUM algorithm to (σ', m', c', t, V).

```
if point of change detected by CUSUM
        → (m', c', t, V)
        Reset RLS:    m → 0 ; c → V ; Pn → P0
        Reset standard deviation: Ns = N – Nd ; Vmean = 0
        Suspend CUSUM test for ηsuspend data points
else
        m' = 0
        c' = 0
        → (m', c', t, V)
end
```

7. Re-calculate record on basis of m' and tank type.

```
If type < 0
        % feeding tank, record = -1 if peak
        If m' > 0
                record = record * sign(m') * -1
        end
end
if type > 0
        % receiving tank, record = -1 if trough
        if m' < 0
                record = record * sign(m')
        end
end
```

8. Dependent on the type of tank, a trough or a peak will indicated that the flow has switched. Keep this time for tool 3B.

```
If type < 0
        % feeding tank, trough detected
        If m' < 0
                t → switch
        end
end

if type > 0
% receiving tank, peak detected
        if m' > 0
                t → switch
        end
end
```

9. Record cycle if record < 0. Feeding tanks - peak to peak. Receiving tanks – trough to trough. When starting to record new cycle, pass previous to Tool 3B.

```
i = 0
If record < 0
        i = i + 1

        if i = 1
                % start tool 3B for previous recorded cycle
                tool3b(cycle,switch)
                clear cycle
        end

        % record the cycle data
        (t, V, ρ) → cycle
else
        % reset record to continue recording
        % switch is 'switching time' of cycle flows
        if type < 0
                % feeding tank
                record = 1 * sign(m') * -1
        end

        if type > 0
                % receiving tank
                record = 1 * sign(m)
        end
end
```

This completes the procedure description of tool 3a. Note that tool 3a technically includes decision 1.

T3.2 Tool 3b Procedure

Tool 3b utilises a simulated annealing algorithm to generate the optimal solutions for the constant flow into, and flow switching times for, the tank. An observer is then used to correct the constant input flow for temporal changes. Because the simulated annealing calculates the average flow rate over a cycle, the average flow rate will be underestimated when temporal changes take place . This underestimation is therefore corrected for.

T3.2.1 Application of simulated annealing

It is recommended that in the first version stages 3 and 4 are replaced with a simple gradient calculation to estimate the constant flow. Stage 5 may be omitted for all versions dependent on if the buffer tank flows estimated by tool 1 are utilised instead.

Tool 3b receives the following data from tool 3a:

cycle' : array containing data points $((t, V, \rho))$
switch : float value for switching time of flows in tank

Tool 3a utilises simulated annealing as an optimisation function. The procedure makes use of the following:

variables:

$t_{c,s}$: cycle start time
$t_{o,f}$: optimisation finish time
$t_{f,g}$: initial flow time estimate
$f_{i,g}$: initial flow in estimate
$f_{o,g}$: initial flow out estimate
$t_{f,o}$: optimal flow time estimate
$f_{i,o}$: optimal flow in estimate
$f_{o,o}$: optimal flow out estimate

1. Set $t_{c,s}$ and $t_{o,f}$. $t_{c,s} = t_1 \in$ cycle'. $t_{o,f}$ = switch
2. Normalise data in cycle' with respect to $t_{c,s}$. $t_i = t_i - t_{c,s}$, $\forall\, t_i \in$ cycle'. Similarly for $t_{o,f}$: $t_{o,f} = t_{o,f} - t_{c,s}$
3. Generate initial guesses for flow in, flow out ($f_{i,g}$,$f_{o,g}$). Generate initial guess for flow on time, ($t_{f,g}$).
4. Run simulated annealing algorithm, generating optimal solution: ($f_{i,o}$, $f_{o,o}$, $t_{f,o}$)
5. Calculate periodic flow using optimal constant flow.

T3.2.2 Estimation of flow out via numerical differentiation

Inputs: **cycle'**: array containing data points ((t, V, ρ)).
($f_{i,o}$, $f_{o,o}$, $t_{f,o}$): optimal solution

Outputs: f_o: flow rate out (array)

The normal tank volume equation can be re-arranged as:

$$f_{out} = f_{in} - \frac{dV}{dt}$$

where the derivative can be differentiated, numerically by applying the symmetric form of the standard equation:

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

to volume measurements with respect to time. For first time point use the standard formula:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

and for the last point:

$$f'(x) = 0.0$$

Hence the procedure is as follows:

1. numerically differentiate V with respect to t → dV
2. for each entry in dV calculate flow out → f_o

T3.2.3 Observer based flow rate correction

$$\frac{dV}{dt} = f_{in} - f_{out}$$

$$\Delta\tilde{f}_{in} = 0.05\left[\hat{V} - V\right]$$

$$f_{in} = \begin{cases} f_{in}(input\_data\_stream) + \Delta\tilde{f}_{in} & , f_{in}(input\_data\_stream) > 0 \\ 0 & , f_{in}(input\_data\_stream) \le 0 \end{cases}$$

T3.2.4 Suppression of Spikes

The flow out estimate from the simulated annealing algorithm is shifted in time from the actual flow out. This shift in time causes spikes in the observer based flow rate correction (see e.g. Figures 12 & 13) because the observer compensates for the mismatch between the flow outs. To suppress the spikes, the observer output is switched to correct flow out $\Delta\tilde{f}_{out}$ instead, for a period of time that encapsulates the estimated flow out. This new value for flow out is then entered in the database.

T3.2.5 Correcting for temporary changes

Inputs: f_{in} : corrected flow rate in (t,f_{in})

Outputs: $\bar{f}_{in}$ new average flow rate (t,f_{in})

The average flow rate will be underestimated when temporal changes and this leads to a raised plateau on the tracking error (Figure 12). To overcome this problem, the corrected flow rate (simulated annealing flow + tracking error) is integrated and then averaged over the entire cycle. Figure 13 shows how this results in the removal of the plateau in Figure 12.

Hence the procedure is as follows:

1. calculate the time interval of the flow in: $\Delta t = t_s - t_f$ where t_s and t_f are the first and last entries in f_{in};
2. integrate f_{in} over the interval t_s to t_f $\int_{t_s}^{t_f} f_{in}dt \rightarrow V_{in}$;
3. calculate the new average flow in: ${V_{in}}/{\Delta t} \rightarrow \bar{f}_{in}$.

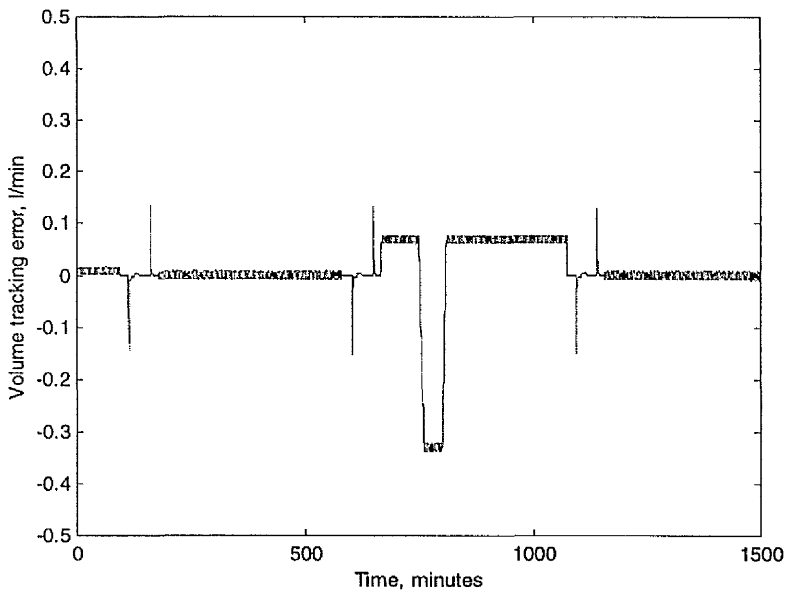


Figure 12: Unadulterated tracking error from volume observer showing raised plateau.

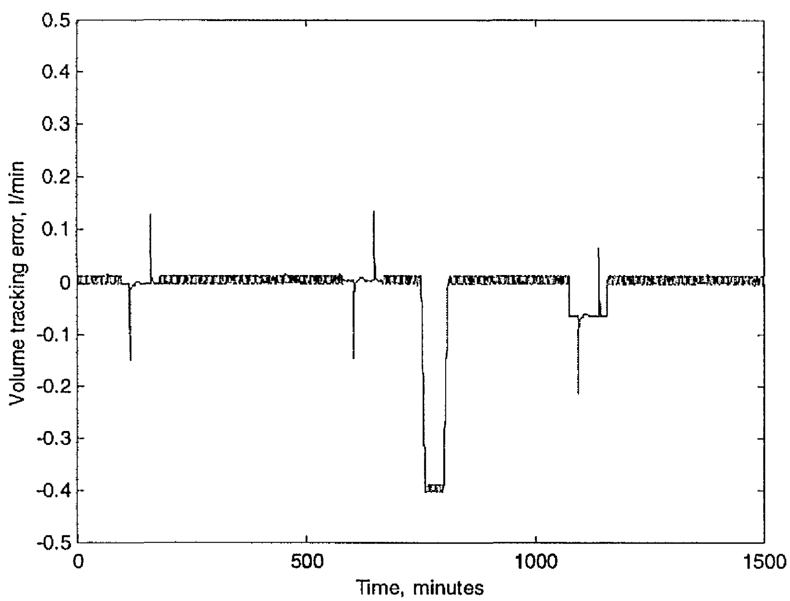


Figure 13: Corrected tracking error using averaged flow rate.

T3.3 Simulated Annealing

The simulated annealing algorithm [24-29] is a method of finding optimum solutions to problems that have a large set of solutions, in an analogous fashion to the physical annealing of solids to attain minimum internal energy states. The fundamental idea is to generate a path through the solution space, from one solution to another nearby solution, leading ultimately to the optimum solution. In generating this path, solutions are chosen from the locality of the preceding solution by a probabilistic function of the improvement gained by this move.

Simulated annealing is very robust since it statistically guarantees finding an optimal solution and is ideal for problems with local minima.

The basic scheme is shown below in Figure 14:

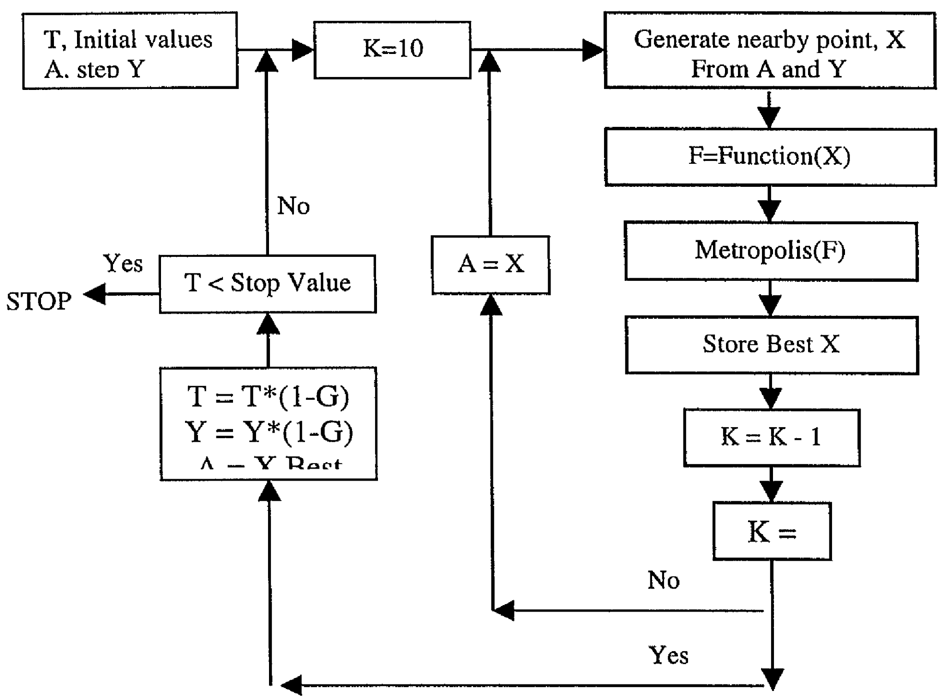


Figure 14: Logical Description of Program

The algorithm is constructed from components. These are:

1. The cooling schedule (manipulation of T).
2. Generation of nearby point.
3. Function to be minimised.
4. Metropolis Function.

T3.3.1 Cooling Schedule

Since the location of the maximum is known to within a few minutes and the flow rates are also known from the previous cycle, the initial value of T can be low as the initial guess is within close proximity of the solution. Thus T = 10.

The reduction in T after each completion of the inner loop is 2.5 percent to ensure an accurate solution is found.

T3.3.2 Generation of nearby point

The purpose is to produce an unbiased walk in the solution space. For each dimension of the solution:

If $R <= 0.5$
$D_{gen} = D + R*G$
else
$D_{gen} = D - R*G$
end

where: D = *old value* D_{gen} = *new value*
$R \in (0,1)$
G = *step length*

The step length G is decreased in a similar fashion as T. This is done so that the search focuses on the region containing the solution.

T3.3.3 Function to be optimised

As stated earlier the function has to have either a maximum or a minimum at the correct values. The simplest method for doing so for this problem is for the function to be a simulation of the volume of the tank. The input arguments for the function are the flow rate in, the flow rate out and the switching time. The function will be of a linear nature. The key to the entire function is the cumulative sum of the differences between the simulated values generated by the function and the measurements taken from the tank. Below are the algorithmic representations of that last statement.

$$Errsum = Errsum + (V_{sim} - V_{meas})^2$$

where:

V = *tank volume*
meas – measurement value *sim – simulated value*
Errsum = cumulative sum of the differences squared

It is obvious that if the function simulated values exactly match the tank measurements then the cumulative sum (Errsum) will be zero.

Given that the above function will have many possible combinations of flow rates and times that will match the data, it was decided to concentrate upon the simulated annealing method so to avoid the problem of local minima.

T3.3.4 Function to be optimised

As stated earlier the function has to have either a maximum or a minimum at the correct values. The simplest method for doing so for this problem is for the function to be a simulation of the volume of the tank. The input arguments for the function are the flow rate in, the flow rate out and the switching time. The function will be of a linear nature. The key to the entire function is the cumulative sum of the differences between the simulated values generated by the function and the measurements taken from the tank. Below are the algorithmic representations of that last statement.

$$Errsum = Errsum + (V_{sim} - V_{meas})^2$$

where:

$V = tank\ volume$

$meas$ – *measurement value* sim – *simulated value*

$Errsum$ = *cumulative sum of the differences squared*

It is obvious that if the function simulated values exactly match the tank measurements then the cumulative sum (Errsum) will be zero.

Given that the above function will have many possible combinations of flow rates and times that will match the data, it was decided to concentrate upon the simulated annealing method so to avoid the problem of local minima.

T3.3.5 Metropolis Function

The Metropolis function [26,29] is the key component in the Simulated Annealing Algorithm as it enables it to avoid local minima.

Have two points in the solution space of function f, 1 & 2. The probability of moving from 1 to 2 is:

$$P(1 \rightarrow 2) = exp(\ -(\ f(2) - f(1)\ )/T)$$

Therefore:

If $f(2) < f(1)$, $P(1 \rightarrow 2) = exp(+ve) = 1$, i.e. downhill changes are always accepted.

If $f(2) > f(1)$, $P(1 \rightarrow 2) = exp(-ve)$, i.e. uphill changes are sometimes accepted but the larger the difference the less likely the change becomes.

Programmatically this can be written as:

```
if R <  exp(-max(F_new - F, 0)/T) then
        x = x_new
        F = F_new
end
```

where: x = *original point* x_{new} = *generated point*

$F = f(x)$ $F_{new} = f(x_{new})$

$R \in (0,1)$ f(a) function to be optimised

TOOLBOX 4: ESTIMATION OF THE X AND ACID COMPONENTS IN THE CONTINUOUS STREAM INTO A RECEIVING TANK.

Inputs:

From calling procedure:

tank_id, t_{start}, t_{finish}, cycle i

From Real-Time Database:

Volume, density, temperature,

f(Input Stream), f(Downstream unit, Stream 1)

[Pu] (process unit upstream, Input Stream)

f (process unit upstream, Input Stream)

Outputs:

To Real-Time Database:

$[X]_{in}$, [Pu], $[H^+]$, ρ_s → Input Stream

[Pu] → Stream 1

f → Process unit upstream, Input Stream

[Pu] → Process unit upstream, Input Stream

I → Process unit upstream

The concentration $[X]_{in}$ is first estimated.

Tool5 is then invoked to estimate the Pu concentration in Stream 1, $[Pu]_{in}$:

Tool 5(Segment_id, t_{start}, t_{finish}, [], [])

Finally the acid concentration $[H^+]_{in}$ is estimated from:

$$[X]_{in} = \alpha_{Pu}(T)*[Pu]_{in} + \alpha_{H^+}(T)*[H^+]_{in} + \ldots + \alpha_{unsp.}(T)*[unsp.]_{in}$$

where α(T) are temperature dependent coefficients and unsp. are all the other (unspecified) components combined (assumed to be zero if 'not known').

T4.1 Estimation of the concentration $[X]_{in}$

Inputs:

From calling procedure: Tank_id

From Real-Time Database:

$f_{in} \equiv$ f(input stream)

V = Volume(Tank_id)

T = Temperature(Tank_id)

ρ_s = Density(Tank_id)

$f_{out} \equiv$ f(downstream unit, Stream 1)

Estimate $[\tilde{X}]_{in}$ is obtained using the observer:

$$[\tilde{X}]_{in} = K_X\left(\hat{V}\bar{\rho}_X - M_X\right)$$
$$\tilde{f}_X = f_{in}[\tilde{X}]_{in}$$
$$\frac{dM_X}{dt} = \tilde{f}_X - f_{out}\bar{\rho}_X$$

where $\bar{\rho}_X$ is the EWMA of ρ_X :

$$\bar{\rho}_X(i+1) = \lambda\rho_X(i) + (1-\lambda)\bar{\rho}_X(i)$$

and ρ_X is obtained from

$$\rho_s = \rho_w(T) + \rho_X$$

T4.2 Pu concentration estimation in Stream 1

Call procedure Tool 5 with

Segment_id = upstream process unit,
Mode = fixed hidden inventory,
Input data streams:
$\forall t$: $t_{start} \le t \le t_{finish}$,
I_{nom} nominal plutonium inventory
f {process unit upstream, Input Stream}
[Pu] {process unit upstream, Input Stream}
Output data streams:
$\forall t$: $t_{start} \le t \le t_{finish}$,
f {formed from Stream 1, Stream 2 & Input Stream}

If f (process unit upstream, Input Stream) or
[Pu](process unit upstream, Input Stream) are 'not known',
then
fill forwards in time in the real-time database with the last known values.
(Process dynamics will ameliorate the effect of errors in this assumption).

TOOLBOX 5: SIMULATIONS.

The plant simulations contain two layers, an outer layer in which the simulation is specified plus an inner layer where the simulation is performed.

Predicted time histories (i.e. data streams) pertaining to two sets of variables are output by each of the modules: states and '*measurement predictions*'. These '*measurement predictions*' are those quantities that will be compared with the Inspector Data or the Operator's Declaration. Each simulation can be executed over any specified period of time from t_{start} to t_{finish}. A base set of initial conditions (ICs) is formed by referring to the real-time database.

T5.1 Outer layer

From calling procedure:	Plant segment ID, t_{start}, t_{finish}, revised_ids, revised_streams
To inner layer:	t_{start}, t_{finish}, ICs, input data streams, output data streams, measurement data streams
To calling procedure:	streams: states & measurement predictions $\forall t$: $t_{start} < t \leq t_{finish}$

The outer layer contains calls to separate modules that simulate the flow of material through each plant segment separately. Prior to the execution of the simulation, appropriate input data streams and output data streams are formed by

- extracting & rationalising the relevant data from the real-time database;
- superimposing any hypothesised scenarios onto the individual data streams.

The relevant data streams for the hypothesised scenarios would be available in event_ids and revised_streams. Both input and output data streams would be formed by taking into account rules governing the merging of Streams1&2 into the input stream. If either f(input stream) or f(Stream 1) are 'not known' then that stream can be calculated on the basis of

$$f(input\,stream) = f(Stream1) + f(Stream2)^{\dagger}$$

where a 'not known' f(Stream 2) would be interpreted as zeros.

The rt database would then be updated accordingly. On the other hand if values for both f(input stream) and f(Stream 1) exist <u>and</u>

$$f(input\,stream) \neq f(Stream1) + f(Stream2)$$

† Assumes that there isn't any shrinkage or expansion when streams merge. This is not a major issue because the simulation is based on mass balances so it is the product of volume with density that matters and density can be corrected to ensure that mass is conserved.

then, a decision would have to be at the time of implementation/commissioning as to which was the most appropriate. If f(input stream) is selected then the input data stream would contain f(input stream) and the output data stream pertaining to the unit upstream would contain a flow rate based on

$$\mathit{flow\ rate} = f(\mathit{input\ stream}) - f(\mathit{Stream2})$$

If f(stream 1) is selected then the input data stream would contain

$$\mathit{flow\ rate} = f(\mathit{Stream}\,1) + f(\mathit{Stream}\,2)$$

and the output data stream pertaining to the unit upstream would contain a flow rate based on f(stream 1).

Other associated variables would then be calculated by referring to a number of relationships. For instance the combined mass flow rate is given by

$$f_{in}\rho_{s_{in}} = f(\mathit{Stream1})\rho_s(\mathit{Stream1}) + f(\mathit{Stream2})\rho_s(\mathit{Stream2})$$

Although volumetric flow rate and density are represented explicitly, it is their products that matter. Thus, if necessary it can be assumed that

$$f_{in} = f(\mathit{Stream1}) + f(\mathit{Stream2})$$

and the corresponding density can be calculated. Variables $[Pu]_{in}$, $[H^+]_{in}$, $[U]_{in}$ and $[unsp]_{in}$ are obtained by merging the two streams, for instance in the case of $[Pu]_{in}$:

$$[Pu]_{in} = \begin{cases} \dfrac{f(\mathit{stream}\ 1)[P_u]_1(\mathit{Stream}\ 1) + f(\mathit{Stream}\ 2)[P_u]_2(\mathit{Stream}\ 2)}{f(\mathit{Stream}\ 1) + f(\mathit{Stream}\ 2)} & ; f(\mathit{Stream}\ 2) \geq 0 \\ [P_u]_1(\mathit{Stream}\ 1) & ; f(\mathit{Stream}\ 2) < 0 \end{cases}$$

In addition *measurement data streams* are formed as required (see below).

The following states might, or might not, be included in the process unit models:

- bulk volume
- mass of acid
- mass of plutonium
- mass of uranium
- mass of unspecified stream 2 components

The formats of the various data streams then conform to the requirements of these process models.

T5.2 The Various Inner Layers

From calling procedure:	t_{start}, t_{finish}, ICs, input data streams, output data streams, measurement data streams
To calling procedure:	states & measurement predictions $\forall t$: $t_{start} < t \leq t_{finish}$

Each inner layer is composed of a differential equation solver that calls one of the module models. Each of the tank set modules is based on the basic tank procedure described next.

T5.3 The Basic Tank Procedure

From calling procedure:

Current time, t.
Current values for up to 5 states (balances): M_{bulk} , M_{Pu} , M_{acid} , M_U, M_{unsp}

(Those states not used would simply be zeroed).

Current time values for the following variables:

$$T,\ f_{in}, \rho_{s_{in}},\ f_{out},\ [Pu]_{in},\ [H^+]_{in},\ [U]_{in},\ [unsp]_{in}$$

where f are volumetric flow rates, [ ] are concentrations in g/l and T is the measured temperature.

To calling procedure:

[Pu], $[H^+]$, [U], [unsp] evaluated at time t

$\frac{dM_{bulk}}{dt}, \frac{dM_{Pu}}{dt}, \frac{dM_{acid}}{dt}, \frac{dM_U}{dt}, \frac{dM_{unsp}}{dt}$ evaluated at time t

The balances are as follows:

$$\frac{dM_{bulk}}{dt} = f_{in}\rho_{s_{in}} - f_{out}\rho_s$$
$$\frac{dM_{Pu}}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu]$$
$$\frac{dM_{acid}}{dt} = f_{in}[H^+]_{in} - f_{out}[H^+]$$
$$\frac{dM_U}{dt} = f_{in}[U]_{in} - f_{out}[U]$$
$$\frac{dM_{unsp}}{dt} = f_{in}[unsp]_{in} - f_{out}[unsp]$$

In conjunction with the temperature, the 5 states enable the calculation of density. The normal equation:

$$\rho_s(T) = \rho_w(T) + \alpha_{Pu}(T) * [Pu] + \alpha_{H+}(T) * [H+] + \ldots..$$

can be re-arranged on the basis of:

$$\rho_s(T) = \rho_w(T) + \frac{\rho_s(T)}{M_{bulk}}\left[\alpha_{Pu}(T) * M_{Pu} + \alpha_{H+}(T) * M_{H^+} + \ldots\right]$$

Then

$$V_{bulk} = \frac{M_{bulk}}{\rho_s}$$

$$[Pu] = \left(\frac{M_{Pu}}{V_{bulk}}\right)$$

$$[H^+] = \left(\frac{M_{acid}}{V_{bulk}}\right)$$

$$[U] = \left(\frac{M_U}{V_{bulk}}\right)$$

$$[unsp] = \left(\frac{M_{unsp}}{V_{bulk}}\right)$$

T5.4 The Tank Sets

At each time step, each of the Tank Sets would be simulated by calling TANK a number of times. For example the sequence for a Tank Set that starts from Cycle 1 outlet, then passes through tanks 4, 5 & 6 before exiting through Cycle 2 would be as follows:

receiving tank:

call TANK with the following arguments:

T {measurement data stream:
extracted from rt database – receiving tank)

f_{in} {input data stream:
formed from Stream 1, Stream 2 & Input Stream}

$\rho_{s_{in}}$ {input data stream:
formed from Stream 1, Stream 2 & Input Stream}

$[Pu]_{in}$ {input data stream:
formed from Stream 1, Stream 2 & Input Stream}

$[H^+]_{in}$ {input data stream:
formed from Stream 1, Stream 2 & Input Stream}

$[unsp]_{in}$ {input data stream:
formed from Stream 1, Stream 2 & Input Stream}

f_{out} {output data stream:
formed from
f(buffer tank, Stream 1), f(buffer tank, Stream 2)
& f(buffer tank, Input Stream)}

buffer tank:

call TANK with the following arguments:

T {measurement data stream:
extracted from rt database – buffer tank}

f_{in} {input data stream:
formed from Stream 1, Stream 2 & Input Stream}

$\rho_{s_{in}}$ {calculated locally:
from receiving tank f_{out}, receiving tank ρ_s & from Stream2}

$[Pu]_{in}$ {calculated locally:
from receiving tank f_{out}, receiving tank [Pu] & from Stream2}

$[H^+]_{in}$ {calculated locally:
from receiving tank f_{out}, receiving tank $[H^+]$ & from Stream2}

$[unsp]_{in}$ {calculated locally:
from receiving tank f_{out}, receiving tank [unsp] & from Stream2}

f_{out} {output data stream:
formed from
f(feeding tank, Stream 1), f(feeding tank, Stream 2)
& f(feeding tank, input stream)}

feeding tank:

call TANK with the following arguments:

T {measurement data stream:
extracted from rt database – feeding tank}

f_{in} {input data stream:
formed from Stream 1, Stream 2 & Input Stream}

$\rho_{s_{in}}$ {calculated locally:
from buffer tank f_{out}, buffer tank ρ_s & from Stream 2}

$[Pu]_{in}$ {calculated locally:
from buffer tank f_{out}, buffer tank [Pu] & from Stream 2}

$[H^+]_{in}$ {calculated locally:
from buffer tank f_{out}, buffer tank $[H^+]$ & from Stream 2}

$[unsp]_{in}$ {calculated locally:
from buffer tank f_{out}, buffer tank [unsp] & from Stream 2}

f_{out} {output data stream:
extracted from rt database – f(downstream unit, Stream 1)}

Output the following measurement predictions for each tank: V_{bulk}, ρ_s

T5.5 Solvent-Extraction Stages

Inputs:

Modes:

A. fixed hidden inventory
B. floating hidden inventory

From calling procedure: t_{start}, t_{finish}

Input data streams:

$\forall t$: $t_{start} \le t \le t_{finish}$,

I_{nom} nominal plutonium inventory
f_{in} {formed from Stream 1, Stream 2 & Input Stream}
$[Pu]_{in}$ {formed from Stream 1, Stream 2 & Input Stream}

Output data streams:

$\forall t$: $t_{start} \le t \le t_{finish}$,

f_{out} {formed from Stream 1, Stream 2 & Input Stream}
$[Pu]_{out}$ {formed from Stream 1, Stream 2 & Input Stream}[1]

Outputs:

To real-time database:

$\forall t$: $t_{start} \le t \le t_{finish}$,

$[Pu]_{out}$ {formed from Stream 1, Stream 2 & Input Stream}[2]
states I, I_{hinv}[3]

To operational history database:

To calling procedure:

The nominal plutonium inventory is steady state so to accommodate dynamics,

$$\tau \frac{dI}{dt} + I = I_{nom}$$

where τ is a time constant obtained during commissioning. (Need only be known approximately). A second inventory, I_{hinv}, is created to accommodate deviations from the nominal value. If a mass balance is now applied to the entire cycle:

$$\frac{d(I + I_{hinv})}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu]_{out}$$

or

$$\frac{dI_{hinv}}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu]_{out} - \frac{dI}{dt}$$

[1] Mode B only
[2] Mode A only
[3] Mode B only

Thus the following equations are solved:

Mode A: equation 1 and $\frac{dI_{hinv}}{dt} = 0 \quad \Rightarrow [Pu]_{out} = \frac{f_{in}[Pu]_{in} - \frac{dI}{dt}}{f_{out}}$;

Mode B: equations 1 & 2.

T5.6 Concentrator

A similar approach is proposed for the concentrator.

TOOLBOX 6: DISAGREEMENT DETECTION.

Detectors are used:

1. to identify disagreements in a single data stream: their general location and approximate time span;
2. to detect disagreements between two data streams (e.g. a simulation and plant measurements): start time, location (usually open-ended i.e. no stop time).

Each time a new disagreement is found, a new sub-event is generated with a unique identifier. Symptoms (e.g. general location and approximate time-span) are then attached: *sub-event_id-symptoms*.

T6.1 Tool 6a: Single stream detection

Detection is achieved using procedure *detect*. This is called by a number of different procedures that focus on different data streams. In each case the results are entered into a sub-event object.

Procedure detect

(sequence) $\rightarrow$ [$\mathbf{t_{start}}$, $\mathbf{t_{stop}}$, **time_profile**]

Inputs: **sequence**
Outputs: t_{start}, t_{stop}, **time_profile**

Array **sequence** is tested for any distinctive deviations. The times at which they start & stop are output as well as their time profiles: t_{start} is rounded down to the nearest 10 minutes and t_{finish} up to the nearest 10 minutes to facilitate comparison. The procedure makes use of the standardised cusum.

Parameters: h – height of vmask
k – angle of arms of vmask
mu – mean of tracking error (zero)
D_U – positive detection tolerance
D_L – lower detection tolerance
rows, times – a dynamic arrays

For every point in sequence (y_i):

1. calculate the standardised variable: $y_i - mu \rightarrow z_i$;
2. calculate the positive and negative cusum for z_i:
   $C^+_i = \max((C^+_{i-1} + z_i - k), 0.0)$
   $C^-_i = \min((C^-_{i-1} + z_i + k), 0.0)$
3. if either the positive or negative cusums exceed their respective tolerances at the i^{th} data point, i.e. $C^+_i > D_U$ or $C^-_i < D_L$, i $\rightarrow$ **rows**; do not reset the corresponding cusum;
4. repeat for all data points;
5. analyse rows: process_points(**rows**) $\rightarrow$ **times** ;
6. search sequence and select flagged sections if **times** does not contain zeros:
   search_sequence(**sequence**, **times**) $\rightarrow$ **time-profiles**.

<u>Procedure process_points</u>

(rows) → times

The input is a column vector of row numbers during which the cusum was alarming. Non-consecutive numbers means the alarm has stopped and another started.

Key variables: **times** – a dynamic array

1. If **rows** contains no data, [0 0] → times
2. Otherwise **rows**(1) → p_start.
3. Process the data in **rows**

```
for ii = 2:size(rows)
        if rows(ii) – rows(ii-1) ≠ 1
                % non-consecutive rows ⇒ new alarm
                [p_start rows(ii-1)] → times
                p_start = rows(ii)
        end
end
```

4. Returns array **times**. Each element in **times** consists of a pair of integers, a start row number and a stop row number for the alarm.

<u>Procedure search-sequence</u>

(sequence, times) → time-profiles

Key variables:
r_s – row at which alarm starts
r_f – row at which alarm stops
t_s – time at which alarm starts
t_f – time at which alarm stops
alarm_data – dynamic array

$\forall$ (r_s r_f) $\in$ **times**:

1. find the corresponding times in **sequence** → t_s and t_f
2. round t_s down to the nearest 10 minutes.
3. modify r_s to equate to t_s
4. round t_f up to the nearest 10 minutes.
5. modify r_f to equate to t_f
6. copy the relevant section from **sequence** into **alarm_data**

```
% counter
p=0

for ii = rs : rf
        % copy the relevant section from sequence to alarm data
        p = p +1;
        alarm_data(p)  = sequence(ii)
end
```

T6.2 Tool 6a-1 Receiving tank input stream or feeding tank output stream

Inputs:

From calling procedure: Tank_id, cycle i, t_{start}, t_{finish}

From Real-Time Database:

if Tank_id ∈ Receiving_tanks,

[X](input stream),

f(input stream),

else

f(downstream unit, stream1).

Outputs:

To Real-Time Database:

To Operational History Database:

To Short-Term Assurances: sub-event_ids

<u>Main procedure</u>

sub-event-id-created = .nil.
sub-event_ids = { }
if Tank_id ∈ Receiving_tanks,
detect([X](input stream)) → [$\mathbf{t_{start}}$, $\mathbf{t_{stop}}$, **magnitudes**]
assign($\mathbf{t_{start}}$, $\mathbf{t_{stop}}$, **magnitudes**, sub-event-id-created, sub-event_ids, 'void')
detect(f(input stream)) → [$\mathbf{t_{start}}$, $\mathbf{t_{stop}}$, **magnitudes**]
assign($\mathbf{t_{start}}$, $\mathbf{t_{stop}}$, **magnitudes**, sub-event-id-created, sub-event_ids, 'void')
else
detect(f(downstream unit, stream1))
assign($\mathbf{t_{start}}$, $\mathbf{t_{stop}}$, **magnitudes**, sub-event-id-created, sub-event_ids, 'void')
end

Procedure assign

($\mathbf{t_{start}}$, $\mathbf{t_{stop}}$, **magnitudes**, sub-event-id-created, sub-event_ids, with)

Assigns the results to a sub-event object. Parameter *with* is used to identify groups of diagnoses.

```
if not_empty(magnitudes)
        ∀ m ∈ magnitudes,
         if sub-event-id-created then
                ∀ se ∈ sub-event_ids,
                 if times involved are in common with se
                        if  not(∃ se::diagnosis_id::sd_id::path-type = with)
                                generate-name&sub-class → diagnosis_id/se::diagnosis_id
                                diagnosis_id ∈ se::diagnoses
                                se::scores → (se::scores 3)
                                generate-name&sub-class → sd_id/se::diagnosis_id::sd_id
                                fill-sub-diagnosis-in-sub-event(m, se::diagnosis_id::sd_id)
                        else
                                se::diagnosis_id whose sd_id::path-type = with → d
                                generate-name&sub-class →
                                                                sd_id/se::diagnosis_id::sd_id
                                [score ∈ se::d::scores that pertains to d] - 1 →
                                                                                se::d::scores
                                fill-sub-diagnosis-in-sub-event (m, d::sd_id )
                        end
                 else
                        create-sub-event → sub-event_id ∈ sub-event_ids
                 end
        else
                        create-sub-event → sub-event_id ∈ sub-event_ids
                        sub-event-id-created = .t.
        end
end
```

Procedure create-sub-event

(t_{start}, t_{stop} , m, sub-event_ids) $\rightarrow$ sub-event_ids

Creates a sub-event.

Produce a name: generate-name $\rightarrow$ sub-event_id
Create object sub-event_id $\in$ sub-event_ids:
description =
diagnosis =
diagnoses = (diagnosis_id)
scores = (1)
symptoms = (symptom_id)

diagnosis_id has a sub-diagnosis sd_id which is a sub-class:

path-type	=	measurement_error
path-id	=	if m pertains to volume then so does path-id (e.g. V01), else ρ**
quantity	=	obtain model prediction from the rt database

symptom_id is a sub-class:
error-in = Depends on the data stream
e.g. if f(input stream) then flow
e.g. if [X](input stream) then X
path-id = Derived on the basis of Tank_id
start-time = t_{start}
stop-time = t_{finish}

T6.3 Tool 6a-2 Hidden inventory

Inputs:

From calling procedure: I_{hinv}, t_{start}, t_{finish}, sub-event_ids

Outputs:

To Real-Time Database:

To Operational History Database:

To Short-Term Assurances:

<u>Main procedure</u> :

sub-event-id-created = .t.
detect(I_{hinv}) → [$\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**]
assign($\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**, sub-event-id-created, sub-event_ids, 'void')

<u>Procedure fill-sub-diagnosis-in-sub-event</u>

(m, sub-event_id::diagnosis_id::sd_id)

sub-event_id::diagnosis_id::sd_id is a sub-class where:
path-type = flow
quantity = m converted to a flow rate
path-id = c**h**

T6.4 Tool 6a-3 Pu/H^+

Inputs:

From calling procedure: Pu, H^+ , t_{start}, t_{finish}, sub-event_ids

Outputs:

To Real-Time Database:
To Operational History Database:
To Short-Term Assurances:

Main procedure :

sub-event-id-created = .t.
detect(Pu) $\rightarrow$ [$\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**]
assign($\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**, sub-event-id-created, sub-event_ids, 'molarity')
detect(H^+) $\rightarrow$ [$\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**]
assign($\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**, sub-event-id-created, sub-event_ids, 'Pu')

NB Where their times overlap, molarity & Pu sub-diagnoses should pertain to the same diagnosis

Procedure fill-sub-diagnosis-in-sub-event

(m, sub-event_id::diagnosis_id::sd_id)

sub_event_id::diagnosis_id:sd_id is a sub-class where:

```
if 'H+' ,
        path-type = 'molarity'
        path-id   = c**h**
        quantity  = m
else if 'Pu' ,
        path-type = 'Pu'
        path-id   = c**h**
        quantity =m
else
        path-type = 'unsp'
        path-id   = c**h**
        quantity =m
end
```

T6.5 Tool 6a-4 Pu/unsp

Inputs:

From calling procedure: I_{hinv}, t_{start}, t_{finish}, sub-event_ids

Main procedure :

sub-event-id-created = .t.
detect(Pu) $\rightarrow$ [$\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**]
assign($\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**, sub-event-id-created, sub-event_ids, ' unsp')
detect(unsp) $\rightarrow$ [$\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**]
assign($\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**, sub-event-id-created, sub-event_ids, 'Pu')

NB Where their times overlap unsp & Pu sub-diagnoses should pertain to the same diagnosis

Procedure fill-sub-diagnosis-in-sub-event

(m, sub-event_id::diagnosis_id)

As for Tool 6a-3

T6.6 Tool 6a-4 Interpreters

Inputs:

From calling procedure: Y, t_{start}, t_{finish}, sub-event_ids

Main procedure :

sub-event-id-created = .t.
detect(Y) $\rightarrow$ [$\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**]
assign($\mathbf{t}_{start}$, $\mathbf{t}_{stop}$, **magnitudes**, sub-event-id-created, sub-event_ids, ' void')

T6.7 Tool 6b: Error-based detection

Detection is achieved using procedure *test*. This is called by a number of different procedures that form an error stream on the basis of different data streams. In each case the results are entered into a sub-event object.

<u>Procedure test</u>

(**error**) → [positive, t_{start}, magnitude]

Array **error** is tested for any distinctive deviations. If positive then positive = .t. else .nil. and the time at which the deviations start is output as well as is their maximum value. The procedure makes use of the standardised cusum, where the mean of the **error** is nominally zero.

Parameters: h – cusum decision interval

1. Form cusum arrays for **error** → C^+ and C^-
2. Test every element of C^+ and C^-:
   positive = .nil.
   if not(positive) & $|C^+| > h$ **or** $|C^-| > h$,
   positive = .t.
   time associated with that particular C^+ or C^- → t
   element of **error** at time t → magnitude
   t_{start} = t – 20 minutes
3. If not(positive) return.
4. Starting at (t_{start} + 20), calculate the gradient for each point. If C^+ alarmed, as soon as gradient <= 0.0, take the magnitude of the previous point from **error**, if C^- alarmed instead, as soon as gradient => 0.0, take the magnitude of the previous point from **error**.

Alternative test: cusum might be overly susceptible to 'spikes', particularly those that might arise because of difficulties in predicting the transients during batch transfers. In which case perhaps

$$\text{alarm if} \int_{t_{start}}^{t_{finish}} |\mathbf{error(t)}|\, dt \geq \frac{\tau}{\left(t_{start} - t_{finish}\right)}$$

might be more suitable. The times here pertain to the simulation and parameter τ is a tolerance.

T6.8 Tool 6b-1 Tank-Sets: first pass

Inputs:
From calling procedure: Segment_id, t_{start}, t_{finish}, measurement-predictions
From Real-Time Database: Reference_vectors pertaining to plant Segment_id
Outputs:
To calling procedure: sub-event_ids

By extracting the appropriate values from the Real-Time Database, form reference_profiles corresponding to the measurement-predictions. Compare and store those that are different over the same period of time. Variable *profile* would have two columns time and the actual values.

```
tp = 9999999 ;        sub-event-id-created = .nil.  ; profile_names = {(dummy m tp) }

∀ profile ∈ measurement-predictions,
        form error_profile = reference_profile - profile
        test(error_profile) → [positive, ts, m]
        if positive
                if tp - ts > - tol (i.e. some tolerance)
                        truncate profile so it only pertains to the time period ts to tfinish
                        (profile m ts) ∈ profile_names
                        if ts: tp - tol > ts > tp + tol
                                tp = min(tp , ts)
                        else
                                remove profile pertaining to tp from profile_names
                                tp = ts
                        end
        end
if profile_names ≠ {(dummy m tp)}
        generate-name → sub-event_id
        Create object sub-event_id ∈ sub-event_ids :
                diagnoses = (diagnosis_id)
                diagnosis_id has a sub-diagnosis sd_id which is a sub-class:
                        error-in = measurement
                        quantity = all profiles in profile_names
                        path_id  = tank_id
        ∀ profile ∈ profile_names & profile ≠ dummy,
                add-symptom-to-sub-event(profile, sub-event_id)
```

Procedure add-symptom-to-sub-event

(profile, sub-event_id)

```
Produce a name:  generate-name → symptom_id
symptom_id ∈ sub-event_id::symptoms :
                symptom_id is a sub-class:
                        error-in = pertains to profile
                        path_id  = pertains to profile
                        start-time = pertains to profile
```

T6.9 Tool 6b-2 Tank-Sets: second pass

Inputs:
    From calling procedure:
        Segment_id, t_{start}, t_{finish}, measurement-predictions, sub-event_id, variables
    From Real-Time Database: Reference_profiles pertaining to plant Segment_id
Outputs:
    To calling procedure: boolean 'success'

By extracting the appropriate values from the Real-Time Database, form reference_profiles corresponding to the measurement-predictions. Then do the following:

```
pass = .t. , success = .nil.
∀ profile ∈ measurement-predictions,
        form error_profile = reference_profile - profile
        test(error_profile) → [positive, ts, m]
        if positive → pass = .nil.
end

if pass
        success = .true.
        Produce a name:  generate-name → diagnosis_id
        ∀ profile ∈ variables,
                add-diagnosis-to-sub-event(profile, sub-event_id::diagnosis_id)
```

Procedure add-diagnosis-to-sub-event

(m, sub-event_id::diagnosis_id)

Produce a name: generate-name → sd_id
Alter object sub-event_id::diagnosis_id :

```
        sub_event_id::diagnosis_id:sd_id is a sub-class where:
                if m pertains to 'H+' ,
                        path-type =  'acid'
                        path_id   =  h**t**
                        quantity  =  m
                else if m pertains to 'Pu' ,
                        path-type =  'Pu'
                        path_id   =  h**t**
                        quantity  =  m
                else if m pertains to 'Transfer To Hinv'
                        path-type =  'flow'
                        path_id   =  t**h**
                        quantity  =  m
                else if m pertains to 'To Transfer In'
                        path-type =  'Transfer In'
                        path_id   =  t-1**t**
                        quantity  =  m
                else if m pertains to 'Substitution with acid'
                        path-type =  'Substitution'
                        path_id   =  t**h**
                        quantity  =  m
```

T6.10 Tool 6b-3 U/Pu Ratios

Inputs:

From Real-Time Database:

[U] and [Pu] at Cycle 1 inlet → $[U]_{in}$, $[Pu]_{in}$

[U] and [Pu] at Cycle 1 outlet → $[U]_{out}$, $[Pu]_{out}$

Outputs:

To Real-Time Database: boolean 'normal' → UPu_separated

For each time record, form the test statistic, x: $$x = \left[\frac{\left(\frac{U}{Pu}\right)_{out} - \left(\frac{U}{Pu}\right)_{in}}{\left(\frac{U}{Pu}\right)_{in}} \right]$$

Filter out transient effects like load changes, **y** = EWMA(**x**)

test(**y**) → [normal, t_s, m]

TOOLBOX 7: MODEL-BASED REASONING

The main aim is to hypothesise events, a list of which would be displayed for the inspector. Each hypothesis would be supported by *symptoms* and *diagnoses*.

The plant can be viewed as a number of sub-plants consisting of tank sets or process units. The operation of the tank sets are observable, directly, and whilst the process units are not. Each of the non-observable sub-plants are viewed as single process units. The underlying concept is based on the fact that non-observable sub-plants are sandwiched between observable sub-plants. Thus if a disagreement in an observable sub-plant can be 'explained' by a material transfer across its external boundary, this material can hide, temporarily, in the connecting non-observable sub-plant. This results in two stages of analysis:

1. how can the material be re-distributed to satisfy the diagnostic criteria?;
2. can this re-distribution be corroborated by supporting evidence?

Re-distribution is examined in two different time-frames, the short term and the medium term. The system is not intended to be used to detect the 'slow drip' (e.g. a diversion of less than say 0.01 significant quantity per hour). Separate re-distribution methods are applied (tools 7a&b), whilst corroboration is handled by Tool 7c.

T7.1 Compartment a: Short-term re-distribution

Re-distributes material along paths using observers. An observer is a particular kind of simulation in which one or more variables are varied so that the simulation predicts specified values. The observer then outputs time profiles pertaining to these manipulated variables. Clearly these variables could be varied forever, so, since any incident is likely to be finite, procedure *tailor* is invoked to truncate each of the profiles.

Observer descriptions follow on from a description of the *tailor* procedure. Switched observers are used here enabling the simulation to be executed, 'as normal', over specified periods of time.

<u>Procedure tailor</u>

[time_profile] → t_{start}, t_{finish}, tailored_time_profile

Find the maximum, then zero all values < 5% of max ? Find times at which the incident starts and stops.

Inputs: time_profile

Outputs: tailored_time_profile

Tool 6a already outputs a time profile of the desired shape except noise might be present at either end of the profile. As the mean of the tracking error is 0.0, it is sufficient at this point to simply zero the first and last data values in time_profile.

1. Set the first value in the time profile to 0.0.
2. Set the last value in the time profile to 0.0

T7.1.1 Solvent extraction cycle observers

Inputs:

From calling procedure:

Segment_id, observer_type, t_{start}, t_{finish}

From Real-Time Database:

see below.

The simulation is a slightly modified version of that in Tool 5. The simulation is executed some $T_{initial}$ minutes before t_{start} to allow the observer to 'settle' before the incident of interest occurs. The procedure call takes the form:

Inputs:

Segment_id = process unit,
observer_type,
t_{start} - $T_{initial}$,
t_{start},
t_{finish},
I_{hinv}, {at time = t_{start} - $T_{initial}$}
Input data streams:
$\forall t$: $t_{start} \leq t \leq t_{finish}$,
I_{nom} nominal plutonium inventory
f_{in} {Input Stream}
$[Pu]_{in}$ {Input Stream}
Output data streams:
$\forall t$: $t_{start} \leq t \leq t_{finish}$,
f_{out} {receiving tank: Stream 1, Stream 2 & Input Stream}
$[Pu]_{out}$ {receiving tank: Stream 1, Stream 2 & Input Stream}
$[X]_{out}$ {receiving tank: Stream 1, Stream 2 & Input Stream}
$[unsp]_{out}${receiving tank: Stream 1, Stream 2 & Input Stream}

Outputs:

To calling procedure:
$\forall t$: $t_{start} \leq t \leq t_{finish}$,
$[Pu]_{out}$ {if Observer_type = Pu/H^+ or Pu/unsp out}
$[H^+]_{out}$ {if Observer_type = Pu/H^+ out}
$[unsp]_{out}${if Observer_type = Pu/unsp out}
state I_{hinv}

The equations are the same as those given in the appropriate part of Tool 5 (Solvent Extraction Stages - Mode B) but with modifications as given below.

Observer_type = Hidden inventory

Execute from t = t_{start} - $T_{initial}$.

If t < t_{start} ,

$$[H^+]_{out} = 0.05\left[I_{hinv}\left(t_{start} - T_{initial}\right) - I_{hinv}\right] \qquad \text{(i.e. observe } H^+\text{)}$$

else

$$[H^+]_{out} = [H^+]_{out} \ at\ t = t_{start}. \qquad \text{(i.e. freeze } H^+\text{)}$$

Then

$$[Pu]_{out} = \frac{[X]_{out} - \alpha_{H^+}(T)*[H^+]_{out} - \alpha_{unsp}(T)*[unsp]_{out}}{\alpha_{Pu}(T)}$$

(i.e. instead of output data stream)

To calling procedure:

tailor(I_{hinv})

Observer_type = Pu/H^+ out

Execute from t = t_{start} - $T_{initial}$.

$$[Pu]_{out} = -0.5\left[I_{hinv}\left(t_{start} - T_{initial}\right) - I_{hinv}\right] \qquad \text{(i.e. observe Pu)}$$

$$[H^+]_{out} = \frac{[X]_{out} - \alpha_{Pu}(T)*[Pu]_{out} - \alpha_{unsp}(T)*[unsp]_{out}}{\alpha_{H^+}(T)}$$

To calling procedure:

tailor($[Pu]_{out}$)
tailor($[H^+]_{out}$)

Observer_type = Pu/unsp out

Execute from t = t_{start} - $T_{initial}$.

$$[Pu]_{out} = -0.5\left[I_{hinv}\left(t_{start} - T_{initial}\right) - I_{hinv}\right] \qquad \text{(i.e. observe Pu)}$$

$$[unsp]_{out} = \frac{[X]_{out} - \alpha_{Pu}(T)*[Pu]_{out} - \alpha_{H^+}(T)*[H^+]_{out}}{\alpha_{unsp}(T)}$$

To calling procedure:

tailor($[Pu]_{out}$)
tailor($[unsp]_{out}$)

Observer_type = Interpret Pu

Execute from $t = t_{start} - T_{initial}$.

If $t < t_{start}$,

$$[H^+]_{out} = 0.05\left[I_{hinv}\left(t_{start} - T_{initial}\right) - I_{hinv}\right] \qquad \text{(i.e. observe H}^+\text{)}$$

else

$$[H^+]_{out} = [H^+]_{out} \ at \ t = t_{start}. \qquad \text{(i.e. freeze H}^+\text{)}$$

Then

$$[Pu]_{out} = \frac{[X]_{out} - \alpha_{H^+}(T)*[H^+]_{out} - \alpha_{unsp}(T)*[unsp]_{out}}{\alpha_{Pu}(T)}$$

To calling procedure:

tailor(Pu_{out})

Observer_type = Interpret H^+

Execute from $t = t_{start} - T_{initial}$.

$$[H^+]_{out} = \frac{[X]_{out} - \alpha_{Pu}(T)*[Pu]_{out} - \alpha_{unsp}(T)*[unsp]_{out}}{\alpha_{H^+}(T)}$$

To calling procedure:

tailor($[H^+]_{out}$)

Observer_type = Interpret unsp

Execute from $t = t_{start} - T_{initial}$.

$$[unsp]_{out} = \frac{[X]_{out} - \alpha_{Pu}(T)*[Pu]_{out} - \alpha_{H^+}(T)*[H^+]_{out}}{\alpha_{unsp}(T)}$$

To calling procedure:

tailor($[unsp]_{out}$)

T7.1.2 Tank set observers

Inputs:

From calling procedure:

Plant segment_id, observer_type, t_{start}, t_{finish}, sub-event_ids

From Real-Time Database:

volume, density $\rightarrow \hat{M}_{bulk}, \hat{\rho}_s$

M_{Pu}, M_{acid} $\rightarrow \hat{M}_{Pu}, \hat{M}_{acid}$; (substitution with acid only)

Outputs:

To calling procedure:

a stream (see individual observers)

The simulation is a slightly modified version of that in Tool 5. The simulation is executed some $T_{initial}$ minutes before t_{start} to allow the observer to 'settle' before the incident of interest occurs. The procedure call is the same as that for Tool 5 but with extra parameters, observer_type and $T_{initial}$. The equations are the same as those given in the appropriate part of Tool 5 but with modifications as given below.

Observer_type = Transfer To Hinv

Execute from $t = t_{start} - T_{initial}$.

If $t \geq t_{start}$,

$$\tilde{Q}_{out} = 0.05\left[V_{bulk} - \hat{V}_{bulk}\right]$$ (i.e. observe V_{bulk})

$$Q_{out} = \begin{cases} \tilde{Q}_{out} & , \tilde{Q}_{out} \geq 0 \\ 0.0 & , \tilde{Q}_{out} < 0 \end{cases}$$

Form vector $\Delta\mathbf{V}_T$ for t: $t_{start} \leq t < t_{finish}$:

$$\Delta\mathbf{V}_T(t) = \int_{t_{start}}^{t} \Delta Q_{out}\, dt$$

Apply one-sided Cusum test to detect & obtain times of incident:

Negative-Cusum($\Delta\mathbf{V}_T$) $\rightarrow [t_s, t_f]$

$$\overline{Q}_{out} = \frac{\Delta V_T(t_{finish})}{(t_f - t_s)}$$

$$f_{out} = f_{out}(output\_data\_stream) + \overline{Q}_{out}$$

$$\overline{Q}_{out} \rightarrow new\_stream$$

end

To calling procedure: tailor(new_stream)

Observer_type = Transfer In

Execute from $t = t_{start} - T_{initial}$.

If $t \geq t_{start}$,

$$\tilde{W}_{in} = 0.05\left[\hat{M}_{bulk} - M_{bulk}\right]$$

(i.e. observe M_{bulk})

$$f_{in} = \begin{cases} f_{in}(input\_data\_stream) + \dfrac{\tilde{W}_{in}}{\rho_{s_{in}}} & , f_{in}(input\_data\_stream) > 0 \\ 0 & , f_{in}(input\_data\_stream) \leq 0 \end{cases}$$

$$f_{in} \rightarrow new\_stream$$

end

To calling procedure:

tailor(new_stream)

<u>Observer_type = Addition of Acid</u>

Execute from $t = t_{start} - T_{initial}$.

If $t \geq t_{start}$,

$$\tilde{W}_{in} = 0.05\left[\hat{M}_{bulk} - M_{bulk}\right]$$

(i.e. observe M_{bulk})

$$W_{in} = \begin{cases} \tilde{W}_{in} & , \tilde{W}_{in} \geq 0 \\ 0.0 & , \tilde{W}_{in} < 0 \end{cases}$$

$$[H^+]_2 = 100 * \left[\hat{\rho}_s - \rho_s\right]$$

(i.e. observe ρ_s)

$$\rho_{s_2} = \rho_w(T) + \alpha_{H^+}(T) * [H^+]_2$$

$$\frac{dM_{bulk}}{dt} = f_{in}\rho_{s_{in}} - f_{out}\rho_s + W_{in}$$

$$\frac{dM_{acid}}{dt} = f_{in}[H^+]_{in} - f_{out}[H^+] + \frac{W_{in}[H^+]_2}{\rho_{s_2}}$$

$$\frac{W_{in}}{\rho_{s_2}}, [H^+]_2 \rightarrow new\_stream(1), new\_stream(2)$$

end

To calling procedure:

tailor(new_stream(1))
tailor(new_stream(2))

Observer_type = Addition of Pu

Execute from $t = t_{start} - T_{initial}$.

If $t \geq t_{start}$,

$$\widetilde{W}_{in} = 0.05\left[\hat{M}_{bulk} - M_{bulk}\right]$$

(i.e. observe M_{bulk})

$$W_{in} = \begin{cases} \widetilde{W}_{in} & , \widetilde{W}_{in} \geq 0 \\ 0.0 & , \widetilde{W}_{in} < 0 \end{cases}$$

$$[Pu]_2 = 100 * [\hat{\rho}_s - \rho_s]$$

(i.e. observe ρ_s)

$$\rho_{s_2} = \rho_w(T) + \alpha_{Pu}(T) * [Pu]_2$$

$$\frac{dM_{bulk}}{dt} = f_{in}\rho_{s_{in}} - f_{out}\rho_s + W_{in}$$

$$\frac{dM_{Pu}}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu] + \frac{W_{in}[Pu]_2}{\rho_{s_2}}$$

$$\frac{W_{in}}{\rho_{s_2}}, [Pu]_2 \rightarrow new\_stream(1), new\_stream(2)$$

end

To calling procedure:

tailor(new_stream(1))
tailor(new_stream(2))

Observer_type = Substitution with acid

Execute from $t = t_{start} - T_{initial}$.

If $t \geq t_{start}$,

$$\tilde{W}_{out} = 0.05\left[\hat{M}_{Pu} - M_{Pu}\right]$$

(i.e. observe M_{Pu})

$$W_{out} = \begin{cases} \tilde{W}_{out} & , \tilde{W}_{out} \geq 0 \\ 0.0 & , \tilde{W}_{out} < 0 \end{cases}$$

$$\left[H^+\right]_{add} = 0.05\left[\hat{M}_{acid} - M_{acid}\right]$$

(i.e. observe M_{acid})

$$\frac{dM_{bulk}}{dt} = f_{in}\rho_{s_{in}} - f_{out}\rho_s$$
$$\frac{dM_{Pu}}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu] - \frac{W_{out}[Pu]}{\rho_s}$$
$$\frac{dM_{acid}}{dt} = f_{in}[H^+]_{in} - f_{out}[H^+] + \frac{W_{out}[H^+]_{add}}{\rho_{s_2}}$$

$$\frac{W_{out}}{\rho_s}, [H^+]_{out} \rightarrow new\_stream(1), new\_stream(2)$$

end

To calling procedure:

tailor(new_stream(1))
tailor(new_stream(2))

Concentrator observers

The solvent extraction cycle observers in the absence of any specific details pertaining to the concentrator.

T7.2 Compartment b: Medium-term assurances

Compartment b contains a number of tools (7b-1, 7b-2, 7b-3, 7b-4 & 7b-5) some of which are co-ordinated by a main procedure Tool 7b. All make use of all of the data available in the Real-Time and Operational History Databases.

T7.2.1 Tool 7b

Inputs:

From calling procedure: $Period_i$, t_{start}, t_{finish}

From Real Time Data base:

$\forall$ tank $\in$ internal_tanks $\cup$ product_accountancy_tank:

V at $t = t_{start}$

V at $t= t_{finish}$

Concentrator:

M_{Pu} at $t = t_{start}$ for each Period

$\forall$ t: $t_{start} \leq t < t_{finish}$, $[Pu]_{out}$

$\forall$ cycle $\in$ solvent extraction cycles,

M_{Pu} at $t = t_{start}$ for each Period

$\forall$ t: $t_{start} \leq t < t_{finish}$, $[Pu]_{out}$

$\forall$ t: $t_{start} \leq t < t_{finish}$

f(tank, input stream) $\rightarrow f_{in}$

f(downstream unit, stream1) $\rightarrow f_{out}$

$\forall$ tanks $\in$ internal_tanks $\cup$ product_accountancy_tank:

$\forall$ t: $t_{start} \leq t < t_{finish}$, [Pu] at time t

From Operational History Database: Table-Flows , Table-Errors

Outputs:

To calling procedure: problem_unit, errout, **redistributions**

A simple plutonium mass balance based simulation (Tool 7b-1) is executed over a time period pertaining to the medium term (t_{start} to t_{finish}) and <u>final</u> inventory predictions are compared with those derived from the databases to obtain the error vector, **e**. If significant errors exist then their associated unit identifier is placed in Units. There are now three possibilities: no significant errors, errors that pertain to a buffer tank, and errors that relate to elsewhere.

Table-Flows: contains a copy of all the flow rate data that pertains to the last 7 periods [i.e. probably 7 * ($t_{start} - t_{finish}$)].

Table-Errors: contains a copy of all the error data that pertains to the last 7 periods, $Period_{i-6}$ to $Period_i$.

Main Procedure

Shift columns of Table-Flows one column left (i.e. lose the oldest column);

Insert current flow rate data (i.e. pertaining to Period_i) into the right column of Table-Flows.

Tool 7b-1 → [Units, **e**]
nul → problem_unit

If ∃unit ∈ Units: unit is the last but one tank in the entire plant simulated,

% problem is in the area of a feeding tank / process unit/ receiving tank
% or the last but one tank

take-samples
Tool7b-3 → Units
unit → starting from the first tank in the plant, then
unit is the last process unit ∈ Units
errout = e(unit identifier that pertains to last but one tank)
τ(unit) = 2*errout
Units → problem_unit

else if ∃unit ∈ Units: unit is the product accountancy tank,

% material in wrong tank

take-samples
Tool7b-3 → Units
unit → starting from the first tank in the plant, then
unit is the last process unit ∈ Units
errout = e(unit identifier that pertains to the product accountancy tank)
τ(unit) = 2*errout
Units → problem_unit

else if Units is not empty,

% problem is with a buffer tank

∀unit ∈ Units, find unit: e_{unit} is a maximum
τ(unit) = 2*e(unit)
errout = e(unit)
unit → problem_unit

end

Tool 7b-2(**e**) → [problem_unit, error, **redistributions]**

∀unit, reset τ(unit)

Procedure take-samples

advise_user("take samples")
wait until advised that this sample data has entered the samples database

T7.2.2 Tool 7b-1 Simulate and compare

Obtain initial conditions pertaining to the start of Period$_{i-6}$ from the Real-Time Database.

For k = 1 to 7,

j = i – 7 + k

1. Execute the simulation over Period$_j$ to obtain $\tilde{M}_{Pu}$ for each unit.
2. $\forall$ unit $\in$ internal_tanks $\cup$ product_accountancy_tank,

   form the error e(unit): $e_{unit} = \hat{V}[Pu] - \tilde{M}_{Pu}$

   where $\hat{V}$ (the volume measured) and [Pu] (the plutonium concentration estimate) are obtained from the real-time database.

   $\forall$ unit $\in$ cycles $\cup$ concentrator:

   form the error e(unit): $e_{unit} = \overline{M}_{Pu} - \tilde{M}_{Pu}$

   where the inventories $\overline{M}_{Pu}$ are obtained from the real-time database (the sums of both the nominal and hidden inventories).
3. Write e_{unit} into the appropriate row of the kth column of Table-Errors.

Test Table-Errors against a specified inventory tolerance for each unit, τ(unit): apply the Cusum test to each unit over 7 periods with h = τ(unit), k = ??

Simulation

$\forall$ units $\in$ internal_tanks $\cup$ product_accountancy_tank $\cup$ cycles $\cup$ concentrator:

$$\frac{dM_{Pu}}{dt} = f_{in}[Pu]_{in} - f_{out}[Pu]$$

$$[Pu] = \frac{M_{Pu}}{V}$$

Obtain the flow rates from the appropriate columns of Table-Flows.

T.7.2.3 Tool 7b-2 Medium Term Re-distribution

A simple procedure then redistributes material along paths on the basis of a number of assumptions:

1. the inventory in each tank or process unit can only be in error by a specified amount;
2. no material can be redistributed upstream from the first tank i.e. material flow measurements from the input accountancy vessel are assumed to be perfect;
3. for similar reasons, no material may be redistributed downstream from the product accountancy tank.

Let ΔM_{unit} denote the incremental plutonium mass redistribution forwards from the unit and let $\Delta M_{unit} = 0$. Starting at the product accountancy tank and ending at the tank immediately downstream of the input accountancy tank:

1. calculate the total error in the tank by summing the error in the tank with the material carried forward:
   $\bar{e}_{unit} = e_{unit} + \Delta M_{unit}$
2. calculate the material carried forward out of the upstream tank/process unit:
   $\Delta M_{unit\ upstream} = \max(0, abs(\bar{e}_{unit}) - \tau_{unit}) * sign(\bar{e}_{unit})$
3. revise the error on the basis of a backward redistribution:
   $\bar{e}_{unit} = \bar{e}_{unit} - \Delta M_{unit\ upstream}$
4. total redistribution forwards from unit, $M_{unit} = \Delta M_{unit}$.

If the first tank is reached and $\Delta M_{unit\ upstream} = 0.0$ then

$[M_{unit1}, M_{unit2}, \ldots] \rightarrow$ **redistributions**

else

redistribute forwards. Starting at the tank immediately downstream of the input accountancy tank and ending at the product accountancy tank:

1. negate the effect of the 'backward' calculation: $\vec{e}_{unit} = \vec{e}_{unit} + \Delta M_{unit\ upstream}$
2. calculate forwards instead, $\Delta M_{unit} = \max(0, abs(\vec{e}_{unit}) - \iota_{unit}) * sign(\vec{e}_{unit})$
3. revise the error on the basis of this redistribution: $\vec{e}_{unit} = \vec{e}_{unit} - \Delta M_{unit}$
4. total redistribution forwards from unit is now, $M_{unit} = M_{unit} - \Delta M_{unit}$

$[M_{unit1}, M_{unit2}, \ldots] \rightarrow$ **redistributions**

end

T7.2.4 Tool 7b-3 Evaluation following on from sampling

Suppose that the following Pu concentrations are available: $\tilde{P}u(Tank_i, t_i)$, $\tilde{P}u(Tank_j, t_j)$, ... Obtain associated Pu concentrations from the real-time database, $\hat{P}u(Tank_i, t_i)$, $\hat{P}u(Tank_j, t_j)$, ..., and form error vector, **error**:

$$error_i = \left|\tilde{P}u(Tank_i, t_i) - \hat{P}u(Tank_i, t_i)\right| ,$$

Working from the first tank forwards identify the first tank, $tank_k$: $error_k$ > accuracy of sample data, then {all tanks/process units upstream from $tank_k$ until, but not including that upstream tank in which a sample has been taken i.e. an $error_j$ exists} $\in$ Units. If Units = { }, then Units = { concentrator feeding tank , concentrator , concentrator receiving tank }.

T7.2.5 Tool 7b-4 Identification of systematic multiplicative errors

This tool is based on the approach devised by Howell & Miller [30].

Inputs:

From calling procedure: t_{start}, t_{finish}

From Real-Time Database:

As for Tool 7b

From Operational History Database: data object Tank-set_id, an instance of Tank-sets

There are a number of preparatory stages that must be carried out before appropriate software can be developed. These involve specifying a number of simulations and deriving qualitative reasoning tables. This information would then be programmed in procedures.

Preparatory stages

Generate tables to relate a qualitative vector, which describes the disagreements, to a set of qualitative vectors that represent possible bias patterns. To do this, identify which tank volumes are to be used in the estimation of which flow rates (e.g. as in Table A7.1) and hence construct the tank-set equations. Form matrix **A** and produce a set of square matrices by eliminating columns. Invert each of these and operate on the disagreement vector to obtain sets of equations that relate the biases to the disagreements. A separate table is now obtained by analysing each set separately. These tables are stored as part of the tank-set_id object i.e. tank-set_id::tables.

Produce a volume-based simulation for each tank-set:

Sim(tank-set_id, t_{start}, t_{finish}) $\rightarrow$ **Volume_estimates**

Produce a simple plant simulation, which propagates [Pu] down the plant (as opposed to uses [Pu] values stored in the Real-time Database):

Sim('overall', t_{start}, t_{finish}) $\rightarrow$ M_{Pu} , [Pu] estimates in the product accountancy tank

The algorithm

∀ tank-set_id ∈ tank-sets,

Sim(tank-set_id, t_{start}, t_{finish}) → **Volume_estimates**
qualitative-compare(**Volume_measurements**, **Volume_estimates**) →
error{tank-set_id} (i.e. a qualitative vector), **V_error**
Table(tank-set_id, **error**) → **BIASES**
Order-bias-combinations(**BIASES**) → **BIAS-COMBINATIONS**

starting with the most likely **bias_combination** ∈ **BIAS-COMBINATIONS**:
if unique-solution?(**bias_combination**) then
least-squares(**bias_combination**) → **bias_values**
if eval-unique(**bias_values**) = .t. then next tank-set_id
else
quantify-bias-combination(tank-set_id, **bias_combination**)
→ **bias_values**
if eval-non-unique(**bias_values**) = .t. then next tank-set_id
try another bias-combination

Procedure qualitative-compare

(tank-set_id, Volume_measurements, Volume_estimates) →
error{tank-set_id} (i.e. a qualitative vector), **V_error**{tank-set_id}

∀ tank ∈ tank-set_id::tanks
d = **volume_measurements(tank) – volume_estimates(tank)**
if |d| < tank-set_id::tolerance
d = 0
end
error(tank) = sign(d)
V_error(tank) = d

The function sign(x) returns +1 if d is positive, -1 if d is negative and 0 is d is zero. This function produces the qualitative vector error{tank-set_id}.

Procedure Table

(tank-set_id, **error**) → biases

Obtain the set **BIASES** from the 'union' of the outputs from the separate tables tank-set_id::tables.

Procedure Order-bias-combinations

(BIASES) → BIAS-COMBINATIONS

By comparing individual elements across the combinations, first choose those combinations with the most common, specified elements, repeat but include unspecified elements specified with their most likely sign, then choose the next most common and so on until all the unspecified elements have been specified with all possible values.

Procedure unique-solution?

(bias_combination)

If the bias contains at least one zero element, then answer is always non-unique.

Procedure least-squares

(bias_combination) → bias_values

Evaluate the biases using the least squares algorithm. Using matrix **A**, remove those columns whose element in **bias_combination** is estimated to be zero. Generate co-variance matrix **D**. Solve using least squares: $[\mathbf{B}^T\mathbf{D}^{-1}\mathbf{B}]^{-1}\mathbf{B}^T\mathbf{D}^{-1}$**V_errors → bias_values.**

Procedure eval-unique

(bias_values)

Re-run both simulations with **bias_values**:

Sim(tank-set_id, **bias_values**, t_{start}, t_{finish}) → **Volume_estimates**

Sim('overall',**bias_values**, t_{start}, t_{finish}) → M_{Pu} ,

qualitative-compare(**Volume_measurements**, **Volume_estimates**) →

error{tank-set_id} (i.e. a qualitative vector), **V_error**{tank-set_id}

If every element in **V_error** < tolerance and MPu matches samples then solution is good.

Procedure quantify-bias-combination

(tank-set_id, **bias_combination**) → **bias_values**

Called when solution is non-unique. The elements of **bias_combination** are repeatedly changed until the Pu concentration in the product accountancy tank correlates with that measured. Establish two bounds to the search:

least-squares($\{\varepsilon_r, 0, 0, \ldots, 0\}$) → **bv_r**

least-squares($\{0, 0, \ldots, 0, \varepsilon_f\}$) → **bv_f**

Based on **bias_combination**, iterate:

[bias_combination(1) + **bias_combination**(last element)**]** = **bv_r**(1) + **b_f**(last element)

until Pu concentration in the product accountancy tank correlates with that measured. Return these valus as **bias_values.**

Procedure eval-non-unique
(tank-set_id, **bias_values**)

Re-run both simulations with **bias_values**:
Sim(tank-set_id, **bias_values**, t_{start}, t_{finish}) → **Volume_estimates**
Sim('overall',**bias_values**, t_{start}, t_{finish}) → M_{Pu} ,
qualitative-compare(**Volume_measurements**, **Volume_estimates**) →
error{tank-set_id} (i.e. a qualitative vector), **V_error**{tank-set_id}
If every element in **V_error** < tank-set_id::tolerance and MPu matches samples then return .t.

T7.2.6 Tool 7b-5 Flow meter calibration

Flow meters installed in the solvent-extraction area apparently [6] have low accuracy but high precision. They could therefore be calibrated from the flow rate data generated by Tool 3 and corroborated by Tool 7b-4.

T7.3 Compartment c: Corroboration and event generation

Inputs:

From calling procedure: sub-event_ids

Individual sub-events would be analysed further to corroborate their diagnoses. Sub-events would then be correlated in time. Event objects would then be generated for the most appropriate. The most appropriate would be chosen on the basis of its *score*. Finally temporal reasoning would be applied to combine events in time. Two procedures invoke various rule-bases:

T7.3.1 tool 7c-1

$\forall$ s $\in$ sub-event_ids: corroborate individual sub-events
correlate sub-events in time
make-events(sub-event_ids)
combine events in time

T7.3.2 Tool7c-2

combine events in time

These tools make use of procedure *confirm.*

Procedure confirm (feature status)

This tool examines the relevant feature in the Real-Time Database to confirm that its activity is as declared by 'status' and returns True or False (see Tool 8). Relevant features include:

- the flow rate of a particular stream;
- the Boolean output of a neutron detector.

T7.3.3 Corroboration of individual sub-event_ids

A rule-base would be provided to guide the corroboration. Being heuristic, rules would be added with operational experience. Examples are given below. Procedures denoted by *(confirm feature status)* access data created by Tool 8 and return a Boolean True/False.

Inputs:

From calling procedure: Sub-event_id

Rules are applied at a diagnosis level

Form an array **sy** containing all the symptoms of sub-event_id

$\forall$ d $\in$ sub-event_id::diagnoses

form an array **sd** containing all the sub-diagnoses of d

form an array **paths** containing all path_id pertaining to **sd**
score is the appropriate element of sub-event_id::scores

if $\exists p \in$ **paths**, which pertains to a solvent-extraction cycle

cycle_id is that cycle that pertains to p
solex-rule-base(**sy**, d, p, cycle_id, score) $\rightarrow$ score
score replaces the appropriate element in sub-event_id::scores

else

if $\exists p \in$ **paths**, which pertains to a tank

tank-rule-base(**sy**, d, p, score) $\rightarrow$ score
score replaces the appropriate element in sub-event_id::scores

end

Solex-rule-base

Example 1

Based on assumption: relevant acid flow meters are accessed

The feed into a receiving tank would consist of an acid stream into which separated products had been added. The flow rate of the acid stream should remain constant unless the operator has changed cycle operation:

If
(confirm (stream-flowrate (cycle_id acid_out)) constant)
then
score = score - 1

Example 2

Based on assumption: neutron-detectors are accessed

If
not(confirm (neutron-detectors cycle_id) normal)

then
(advise 'abnormal_operation' cycle_id)
score = score + 1

Example 3

Based on assumption: XRF-detectors are accessed

If

    not(confirm (XRF cycle_id) normal)

then

    (advise 'abnormal_operation' cycle_id)
    score = score + 1

Example 4

if

    the time periods of any of the symptoms ∈ **sy** overlap a time period in declared-operational::cycle_id in which the status changed to 'loaded'

&   ∃ s ∈ **sd:** s::path-id = 'c**h**' or 'h**c**' (i.e. pertaining to cycle_id)

then

    score = score + 2

Tank-rule-base

Example 5

if

    the time periods of any of the symptoms ∈ **sy** overlap a time period in declared-operational::tank_id in which the status is 'shut-down'

&   ∃ s ∈ **sd:** s::path-id = 't**h**' (i.e. pertaining to tank_id)

then

    score = score + 2 ?

T7.3.4 Correlate sub-events in time

Example 6
A problem exists with Tool 3. A diversion during normal filling of receiving tank (tank_id) would generate two sub-events because the simulated annealing approach 'smooths' the effect. Correlate in time to combine these.

Obtain from operational database: tank_id::rtank::empty_times $\rightarrow$ ((t_{s1} t_{f1}) (t_{s2} t_{f2})....)

$\exists$ se1, se2 $\in$ sub-event_ids:
  a symptom s1 of se1::symptoms has
    path-id = p1,
    error-in = 'flow',
    start-time = t1 &
    stop-time = t2
  a symptom s2 of se2::symptoms has
    path-id = p2,
    error-in = 'flow',
    start-time = t3 &
    stop-time = t4,
  t1 < t3,
  (t3 t4) $\in$ ((t_{s1} t_{f1}) (t_{s2} t_{f2})....),
then
  se1::link-with = se2
  se1::link-type = 'primary-effect'
  se2::link-with = se1
  se2::link-type = 'secondary-effect'
end

T7.3.5 Make Events

Remove all sub-events with link-type = 'secondary-effect' from sub-events
Group the remaining sub-events into sub-sets:

if two or more sub-events are linked (i.e. via link-with) and have link-type = 'primary-effect'
then put in the same sub-set
else put in separate sub-sets

$\forall$ sub-set: make-event(sub-set) $\rightarrow$ event_id $\in$ event_ids

Procedure make-event

(sub-event_ids) $\rightarrow$ event_id

The most likely sub-event/diagnosis would be chosen by looking through all the scores of all the sub-events $\in$ sub-event_ids for that diagnosis with the largest score:

$\forall$ s $\in$ sub-event_ids,
find that diagnosis, d, that pertains to max(s::scores)
s::diagnosis = d
s::score = max(s::scores)

Produce a name: generate-name $\rightarrow$ event_id
Create object event_id $\in$ event_ids:
description = name on the basis of path-id & path-type
sub-events = (sub-event_ids)

T7.3.6 Combining events in time

Example 7

If

$\exists$event1: event_description = "solex_hinv" ,
event1::path_id = x,
quantity = $\mathbf{q_1}$
start_time $\approx t_1$, stop_time $\approx t_2$.

and

$\exists$event2: event_description = "solex_hinv" ,
event2::path_id = x,
quantity = $\mathbf{q_2}$
start_time $\approx t_3$, stop_time $\approx t_4$.

and

$|t2 - t3| < \textit{some tolerance},$

and

$$\left| \sum_{i=t_1}^{i=t_2} q_{1_i} - \sum_{i=t_3}^{i=t_4} q_{2_i} \right| < \textit{some tolerance}$$

then

(combine event1 event2)

TOOLBOX 8: CONFIRMATION OF OPERATIONAL UNIT STATUSES

Certain measurements are continuously monitored to detect any change in operation: I (increasing), S (steady) or D (decreasing) are recorded in the Real-Time Database. Although a single tool is discussed here, a number of tools tailored for separate data streams would be more likely.

Input: data point $d_n \in \mathbf{d}$ at time $t = \frac{n}{f}$

Outputs: status at time t

Parameters:

μ	mean of variable (initially the instrument output)
k	angle of arms of vmask
D_U	positive detection tolerance
D_L	negative detection tolerance
f	instrument output frequency (outputs/minute)

A detection procedure is positioned in between the instrument output and the real-time database. This procedure makes use of the standardised cusum.

Each time a new point arrives:

calculate the standardised variable $(d_n - \mu) \rightarrow y_i$

calculate the positive and negative cusum for y_i :

$$C_i^+ = \max\left(\left(C_{i-1}^+ + y_i - k\right), 0.0\right)$$

$$C_i^- = \min\left(\left(C_{i-1}^- + y_i + k\right), 0.0\right)$$

If $C_i^+ > D_U$

'I' $\rightarrow$ status

else if $C_i^- < D_L$

'D' $\rightarrow$ status

else

'S' $\rightarrow$ status

If $C_i^+ > D_U$ or $C_i^- > D_U$,

feed_change = search_data(i,t_i,y_i,**y**),

reset the cusum $\rightarrow$ $C_i^+ = 0$ & $C_i^- = 0$, $\mu = y_i$.

Procedure search_data(i,t_i,y_i, **y**)

Finds the approximate start and stop times for each change. Start time is approximated to ten minutes before change detected, stop time, ten minutes after change detected.

Inputs: i, t_i, y_i, and **y**

Outputs: y_s, y_f

Parameters: s – points to start
f – points to stop
t_s – time at which section starts
t_f – time at which section stops

Calculate start and stop times: $t_s = t_i - 10.0$, $t_f = t_i + 10.0$.
Round t_s and t_f to the nearest 10 minutes.
Calculate s to equate to t_s
Calculate f to equate to t_f

Appendix 6: Case studies

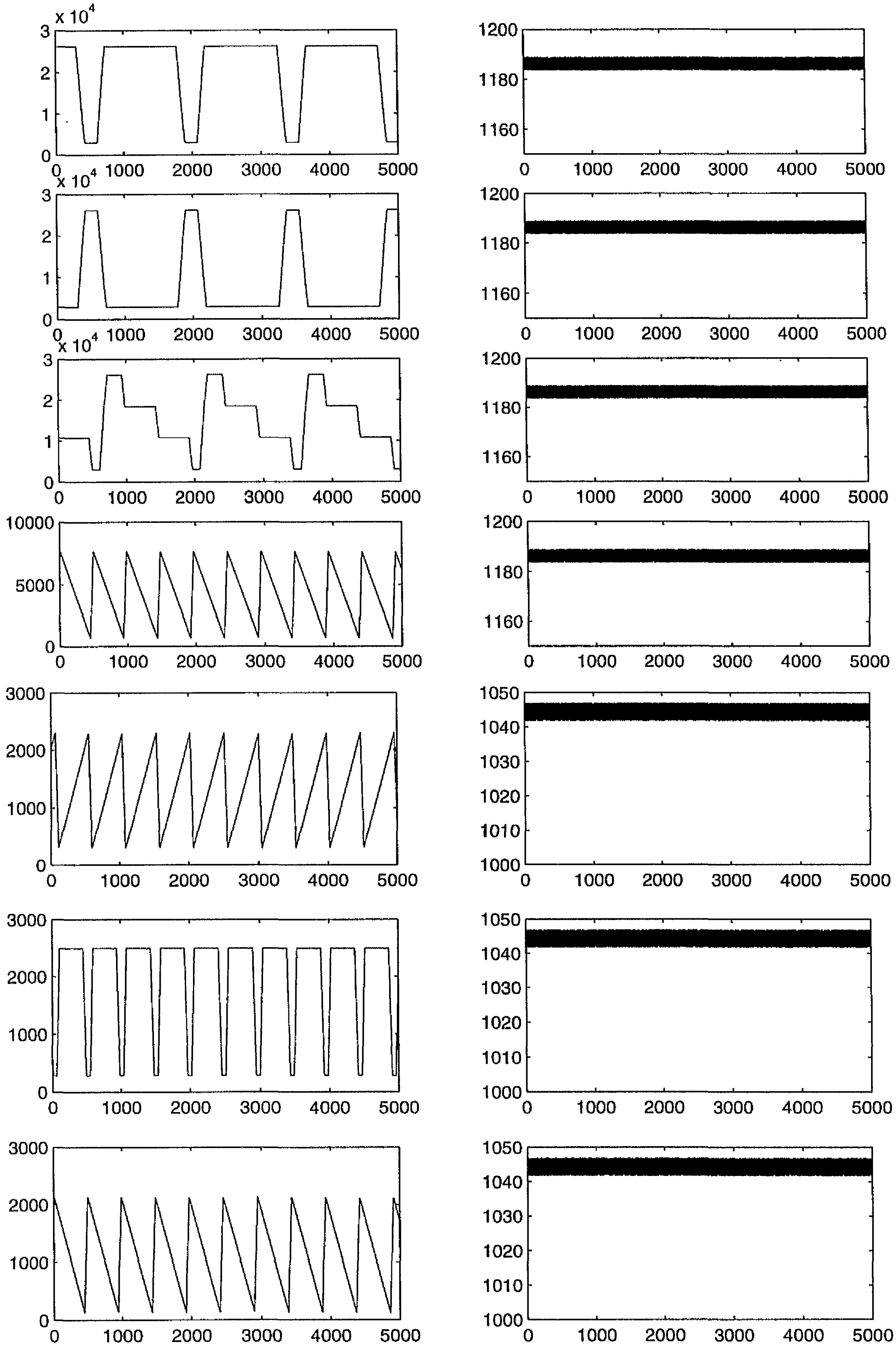
A6.1 Introduction

In producing these examples, our remit was to examine overall performance without spending too much time on methods development. The computer simulation on which these results are based was crude. The procedures that were applied, were not optimised to reflect the fact that real plant data would look different. The results presented below should be viewed with this understanding in mind.

The simulation was of a reprocessing plant with two cycles and a concentrator and hence had the following segments:

1. tank-set 1 consisting of four tanks after the input accountancy tank
2. cycle 1
3. tank-set 2 consisting of three tanks
4. cycle 2
5. tank-set 3 consisting of three tanks
6. concentrator
7. tank-set 4 consisting of two tanks, the second of which is the product accountancy tank.

Figure 15 gives volume and density histories for each tank over the four days covered by the simulation, and for the 'base' case. Thus the first four rows pertain to Tank-set 1 i.e. Tanks 1-4, the next three pertain to Tank-set 2 i.e. Tanks 5-7 and so on. If Tool 7b was to be applied to this data then the table of errors, generated by comparing the simulation with plant data, would look something like that shown in Table 6, and the re-distribution (after four days) would look something like that shown in Table 7.


x 10^4
0
1
2
3
1000
2000
3000
4000
5000
1160
1180
1200
10000
5000
1000
1010
1020
1030
1040
1050

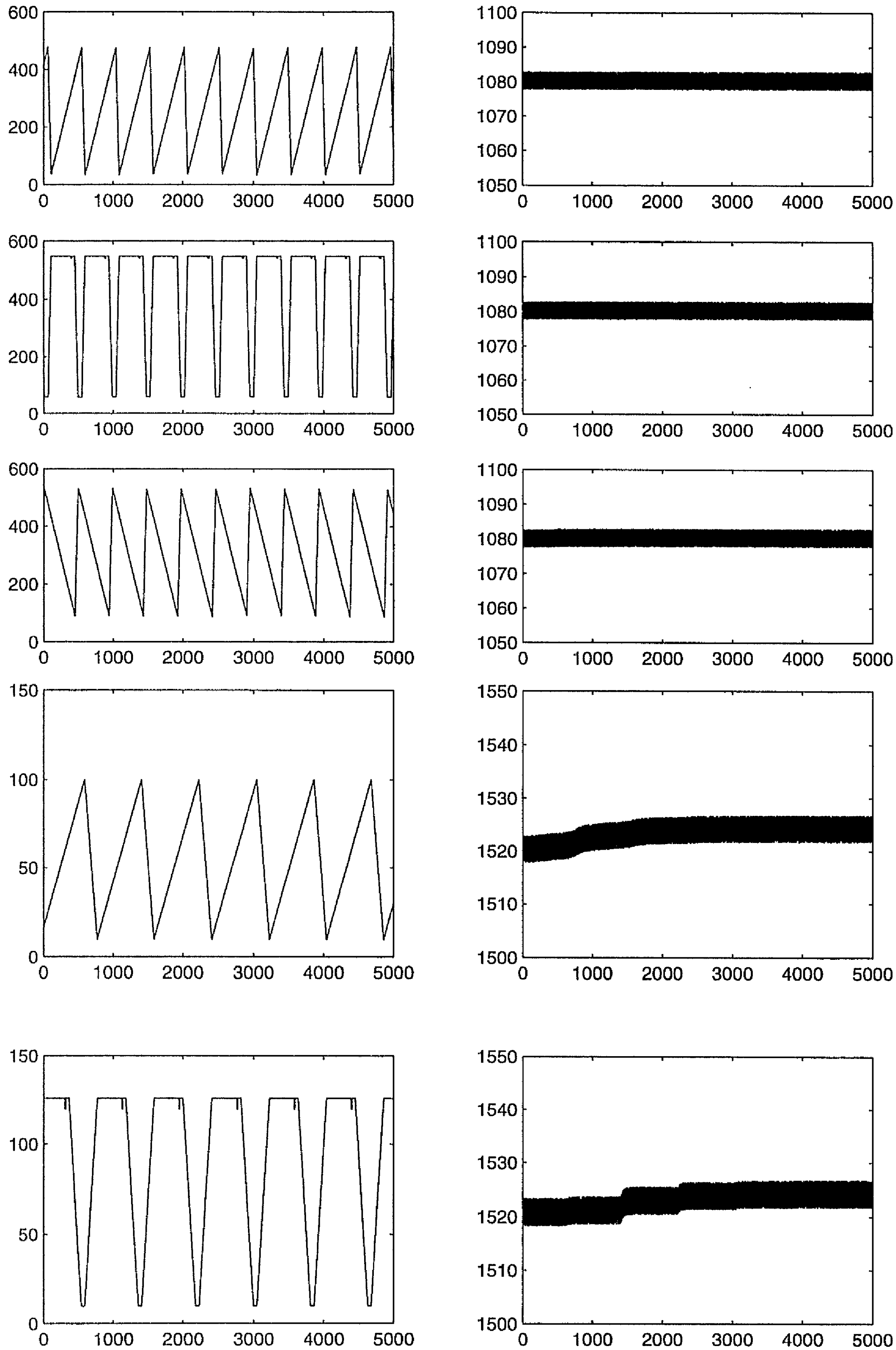


Figure 15: Volume (litres) and density transients (g/l) in all tanks over 5000 minutes.

	1 day(0-1500)	2 days (0-2600)	3 days (0-4400)	4 days (0-5500)
Tank 1	0.01	0.21	0.65	0.71
Tank 2	-9.60	-9.81	-9.98	-9.96
Tank 3	-0.04	-0.01	-0.37	-0.47
Tank 4	-6.51	-6.41	-8.33	-8.76
Solex 1	-0.01	-0.01	-0.02	-0.01
Tank 5	-1.19	0.39	-4.13	-3.46
Tank 6	1.50	0.73	0.26	-0.06
Tank 7	-7.26	-6.26	-4.84	-4.75
Solex 2	0.19	0.21	0.06	0.06
Tank 8	-38.99	-43.28	-75.09	-118.18
Tank 9	1.09	1.88	-5.75	-9.06
Tank 10	18.16	86.10	142.47	191.05
Concentrator	-0.75	-2.27	-1.24	3.14
Tank 11	1.56	10.80	18.04	22.60
Tank 12	-51.44	-51.64	-150.71	-61.99

Table 6: showing errors increasing over 4 days

	Error	Redistribution
Tank 1	0.71	0.71
Tank 2	-9.96	-9.96
Tank 3	-0.47	-0.47
Tank 4	-8.76	-8.76
Solex 1	-0.01	-0.01
Tank 5	-3.46	-3.46
Tank 6	-0.06	-0.06
Tank 7	-4.75	-4.75
Solex 2	0.06	0.06
Tank 8	-118.18	-36.19
Tank 9	-9.06	50.00
Tank 10	191.05	50.00
Concentrator	3.14	3.14
Tank 11	22.60	10.61
Tank 12	-61.99	-50.00

Table 7: Re-distribution If Performed Once At End Of 4 Days

A.6.2 Case 1: Abrupt diversion from the Cycle 2 outlet

This section describes the procedures that would ensue if 20 litre of 30 g/l Pu concentration solution was to be 'taken', at a constant rate over 60 minutes, from the Cycle 2 outlet. It is assumed that the molarity remains constant at the exit of this cycle. This movement is characterised by a temporary reduction in liquor entering the receiving tank (Tank 8).

Figures 16 & 17 show the level and density plots that might be observed: note the incident. This data would be entered into the Real-time Database and analysed by Tools 3&4 (Figures 17 & 18) that would result in the output of flow-rate data to the database and trigger the sequence below:

Tank Set 3, Tool 6

Figures 18 & 19 show an irregularity for about 60 minutes from about 1600 minutes. This is detected by Tool 6 (Figures 20-22) on the basis of the spike shown in Figure 22. The 'start' and 'stop' times and time history pertaining to the flow rate are extracted automatically and a sub-event is created to represent the story so far.

Tool 7a

Three observers are now invoked 'in-parallel': one looks at the possibility of a transfer to hidden inventory (Figure 23), one looks at the possibility of a change in Pu concentration and acid molarity (Figures 24 & 25), simultaneously and one performs a similar function with Pu concentration and Unspecified (Figures 26 & 27). (Simultaneous actions might be needed to satisfy both volume and density measurements).

Tool 6

The various deviations are now detected and scores are attached to each of the three possibilities, which are entered as sub-event diagnoses. The movement to hidden inventory scores highest because it involves only one deviation. Note also that it has estimated the total movement to be about 600 gms.

Tool 7c

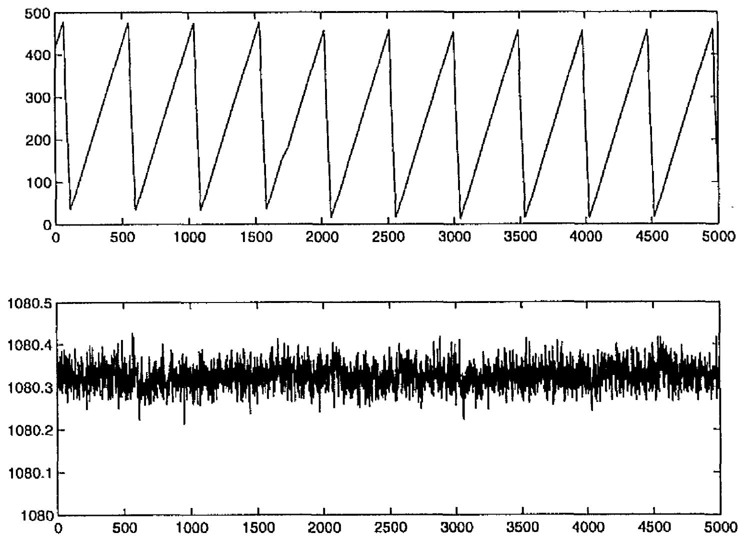
Corroborates the favoured diagnosis by confirming that the flow rate of the relevant inactive feed didn't change during that period. Also that the cycle appears to be operating normally because the neutron detectors and XRF measurements are normal. If the health of the dip-tube system is also okay then

⇒ conclude that there is a disagreement from 1600 – 1660 minutes
⇒ create an event to describe this in the Operational History Database.

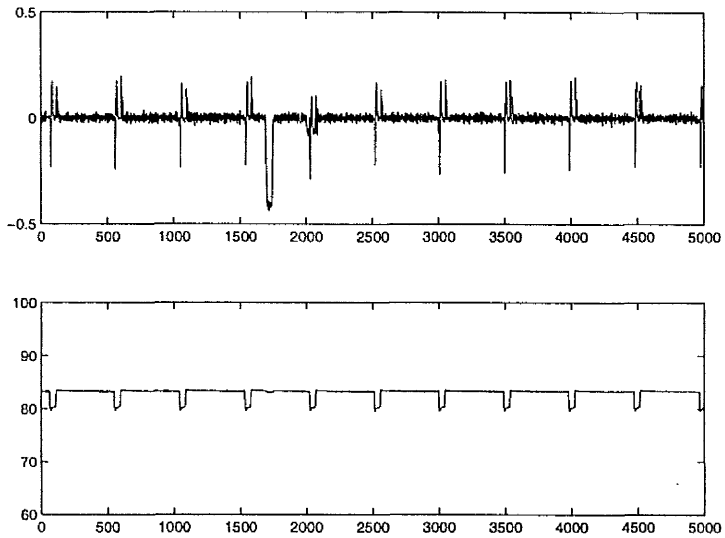
The alternative explanations, either as generated by the other observers or pertaining to the fact that the volume sensor could have developed a bias over the hour, are still available in the Operational History Database if required.

Follow-up:

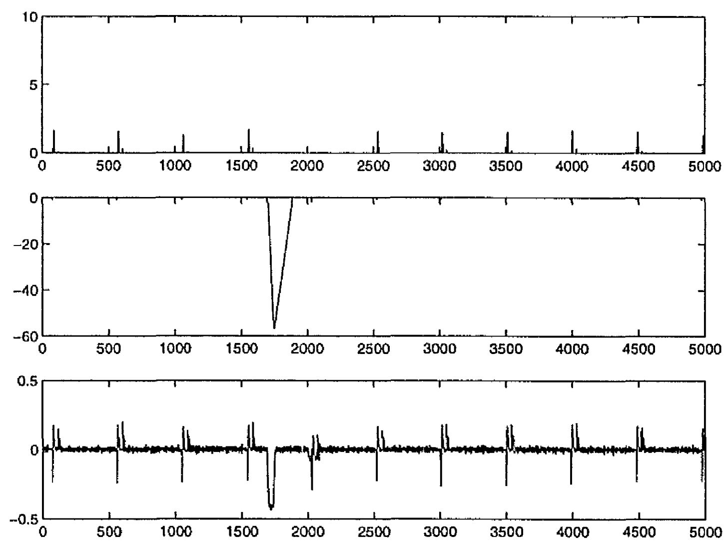
Check to see if anything happened in Cycle 2.
If the dip-tube system does not have a health indicator, verify the performance of the sensor output by comparing the various dip-tube pressure signals.



Figures 16 & 17: Tank 8 volume (litres) & density (g/l)



Figures 18 & 19: Tank 8 observer outputs: flow rate error (l/min) and X (g/l)



Top: positive test signal (detected if | signal | > some threshold)
Middle: negative test signal (detected if | signal | > some threshold)
Bottom: input signal

Figures 20 - 22: Tank 8 detectors

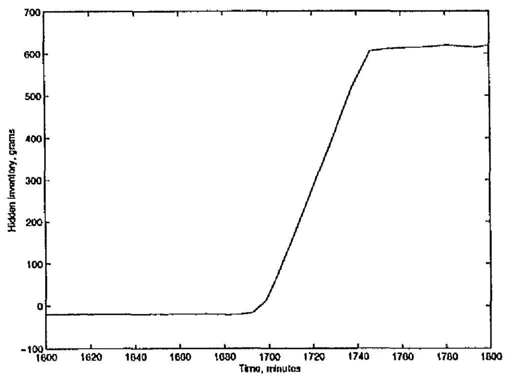


Figure 23: Hidden inventory using observer_type = Hidden inventory

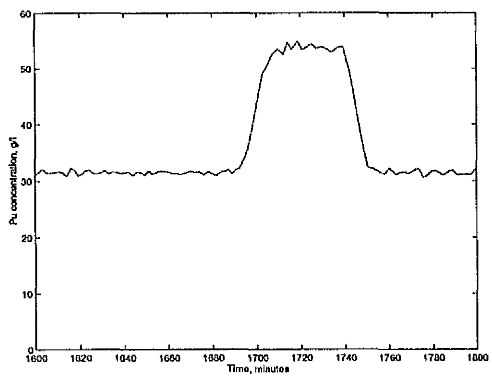


Figure 24: Pu concentration using observer_type = Pu/Acid

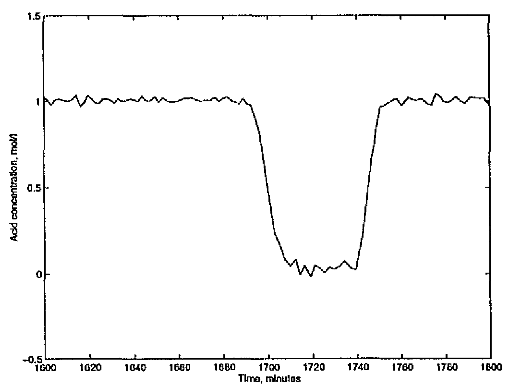


Figure 25: Acid molarity using observer_type = Pu/Acid

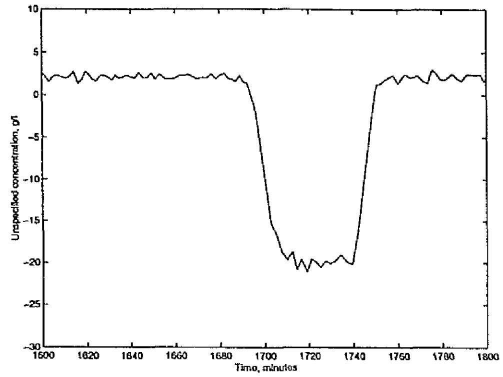


Figure 26: Unspecified using observer_type = Pu/unsp

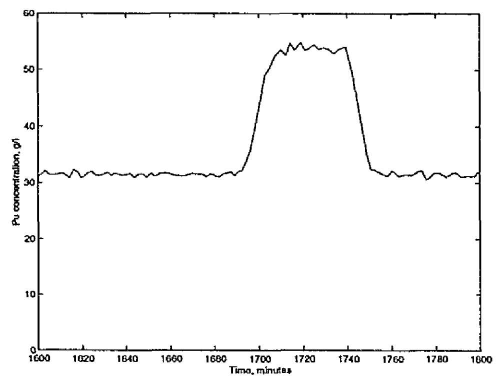


Figure 27: Pu concentration using observer_type = Pu/unsp

A.6.3 Case 2: Temporary increase in Cycle 2 holdup

This section describes the procedures that would ensue if about 600 gms of plutonium was to 'disappear' for a period of about 12 hours, then re-appear (Figure 28). This movement is characterised by a reduction in the density of the liquor in the receiving tank. It is difficult to attribute fluctuations in density with fluctuations in plutonium concentration because density might also fluctuate as a result of changes in acid molarity and in the concentrations of other components. As a consequence of this detection must be relatively insensitive and diagnosis must accommodate the various possibilities.

Tank Set 3, Tool 4

Tool 4 would be applied every receiving tank fill/empty cycle. Figure 29 shows what would be obtained: a pair of symmetrical irregularities, the first, a negative spike, the second a positive spike.

Tank Set 3, Tool 6a

The tolerances in Tool 6a would be set sufficiently large so that it ignores normal fluctuations in density. If these spikes were to be detected then the 'start' and 'stop' times and time history pertaining to the two spikes would be extracted automatically and a sub-event would be created to represent each spike separately.

Tool 7a

Three interpreters are now invoked 'in-parallel': one determines that change in Pu concentration that would cause the variation (Figure 30), one determines the change in acid molarity (Figure 31) and one the change in 'unspecified' (Figure 32).

Tool 6a

The various deviations are now detected and descriptions & scores are attached to each of the three possibilities as part of their sub-event diagnoses. The scores are based on what is most likely: a change in Pu concentration will score highest if the spikes are large enough.

Tool 7c

Initially two separate events are created on the basis of the highest scores. They are then combined together to produce an event ('temporary increase in Cycle 2 hold-up'), which requires no follow-up.

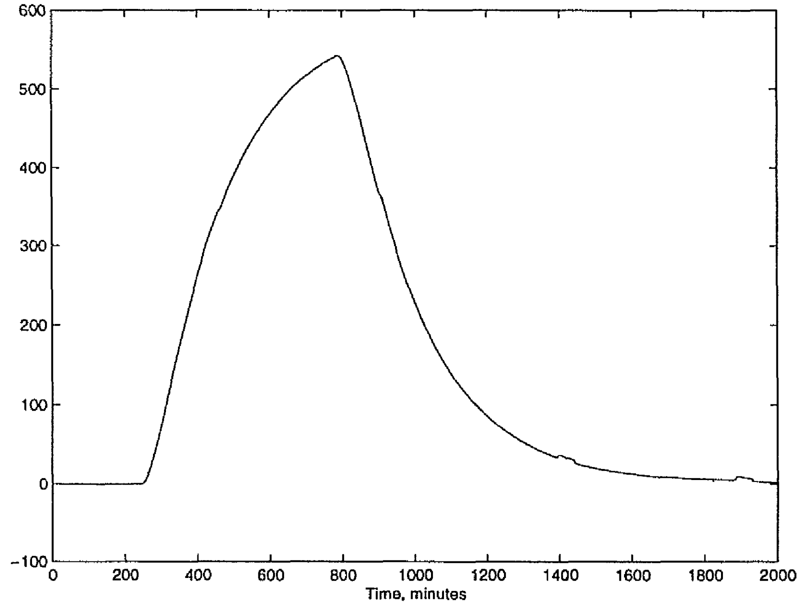


Figure 28: Transient change in Cycle 2 hold-up of plutonium (gms)

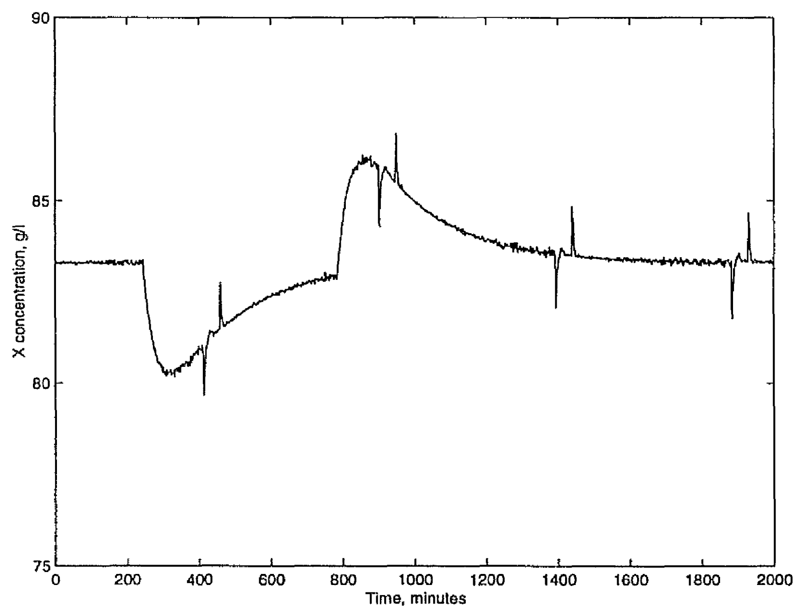


Figure 29: X-Observer Output For Cycle 2 Receiving Tank

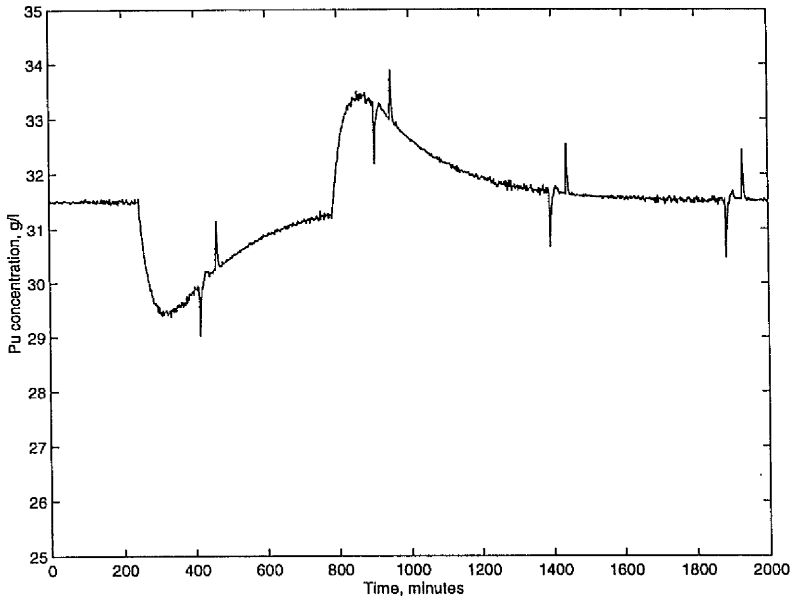


Figure 30: Variation in Pu Concentration Required

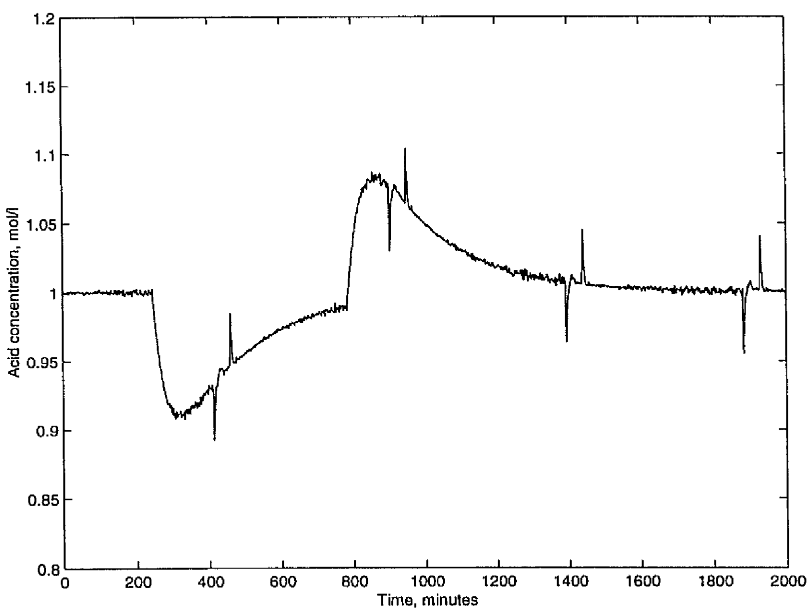


Figure 31: Variation in Acid Molarity Required

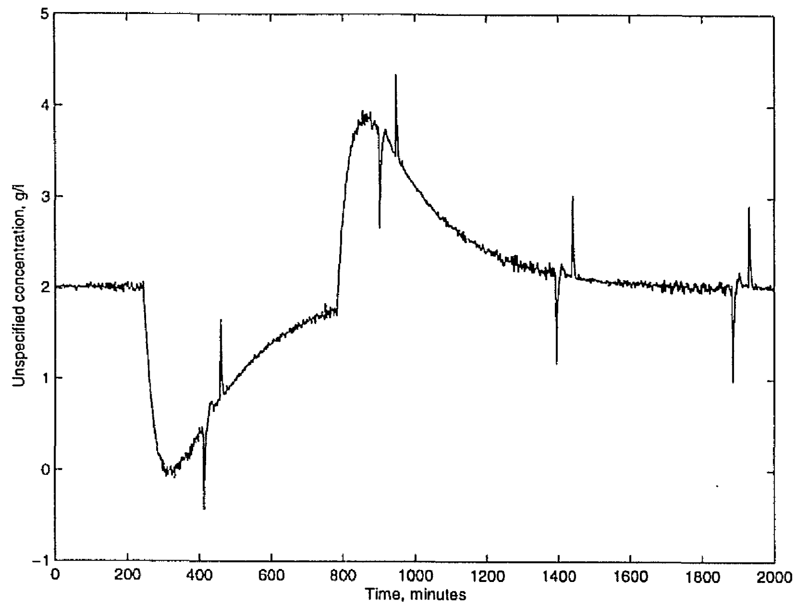


Figure 32: Variation in Unsp Concentration Required

A.6.4 Case 3: Abrupt diversion from Tank 8 during export

This section describes the procedures that would ensue if 20 litre of 30 g/l Pu concentration solution was to be 'taken', at a constant rate, over 15 minutes from the receiving tank located in Tank Set 3, and whilst the tank is being emptied. It is assumed that the molarity remains constant at the exit of this cycle. Depending on which level data is used to estimate transfers between the receiving and buffer tanks, either the movement would be characterised by a reduction in liquor transferred to the batch tank, Tank 9, or in an increase in liquor left in Tank 8. Either situation would be detected by Tool 6b.

<u>Tank Set 3, Tool 3</u>

Figure 33 shows the observer output. Nothing would be detected by Tool 6a in this signal.

<u>Tank Set 3, Tool 5</u>

Figure 34 shows the level plot that might be observed in Tank 8 together with its estimate, where the measurement is the dashed line. Note the divergence from about 940 minutes.

<u>Tool 6b</u>

Figures 35 - 37 show the error signal that is input into the detector together with the two CUSUM (boolean) outputs. A 'start' time at about 920 (minutes) and time history starting from about 900 (minutes) are extracted automatically and a sub-event is created to represent the story so far.

<u>Tool 7a</u>

Up to four observers would now be invoked 'in-parallel': the first looks at the possibility of a transfer <u>out</u> to hidden inventory, whilst the other three look at the possibilities of various forms of addition: plutonium nitrate solution as per the tank upstream, plutonium nitrate solution of a different concentration or acid. In each case a material movement is obtained. Since material is known to be leaving, here only the first would be meaningful (Figure 38). Note the 'spikes' that 'confuse' the picture. These result from the likely temporal mismatch between simulation and reality during the start and finish of batch transfers. A simple procedure is used to extract the average flow rate out, including its start and stop times, from the plot. This is based on the integral of the flow rate (Figure 39), which has estimated the total movement to be about 20 l.

<u>Tool 5</u>

The effect of making this movement is now examined by incorporating each one in turn into the simulation.

<u>Tool 6</u>

The effects are tested and the successful movement is entered as a sub-event diagnosis.

<u>Tool 7c</u>

An event is created to describe this in the Operational History Database.

<u>Tool 7b</u>

It is also worth pointing out that, depending on tolerances, and if the event has not been accommodated in the simulation before Tool 7b is invoked, Table ** might be output by Tool

7b. Note that Tank 8 is suspected.

	Error
Tank 1	0.16
Tank 2	-9.64
Tank 3	-0.16
Tank 4	-5.30
Solex 1	0.00
Tank 5	-1.88
Tank 6	0.29
Tank 7	-6.30
Solex 2	-0.17
Tank 8	607.24
Tank 9	3.83
Tank 10	-4.66
Concentrator	0.49
Tank 11	4.54
Tank 12	-76.98

Table 8: Out flow diversion on tank 8, single cycle simulation

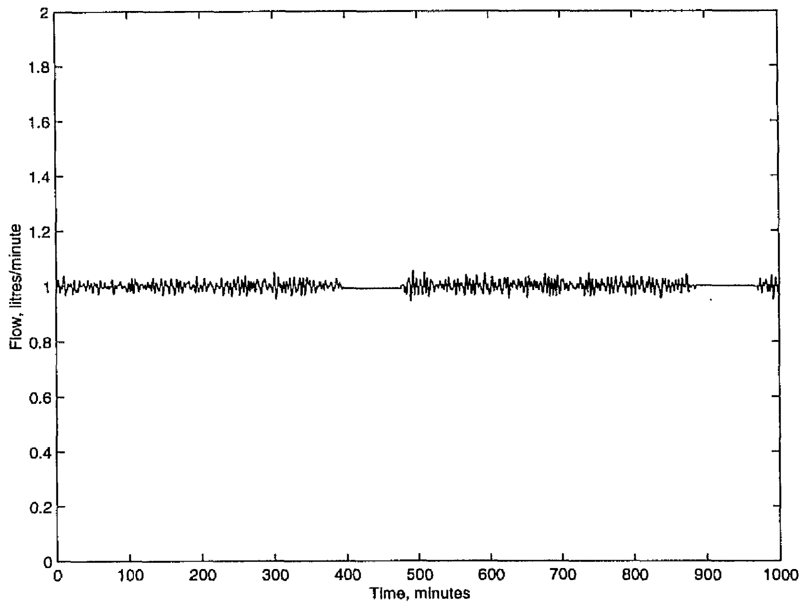


Figure 33: Tool 3 Observer Output For Tank 8

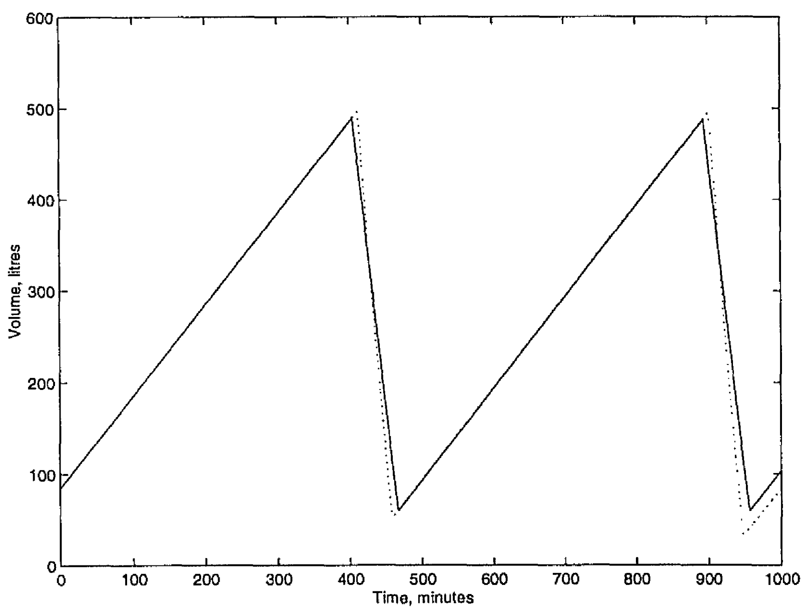
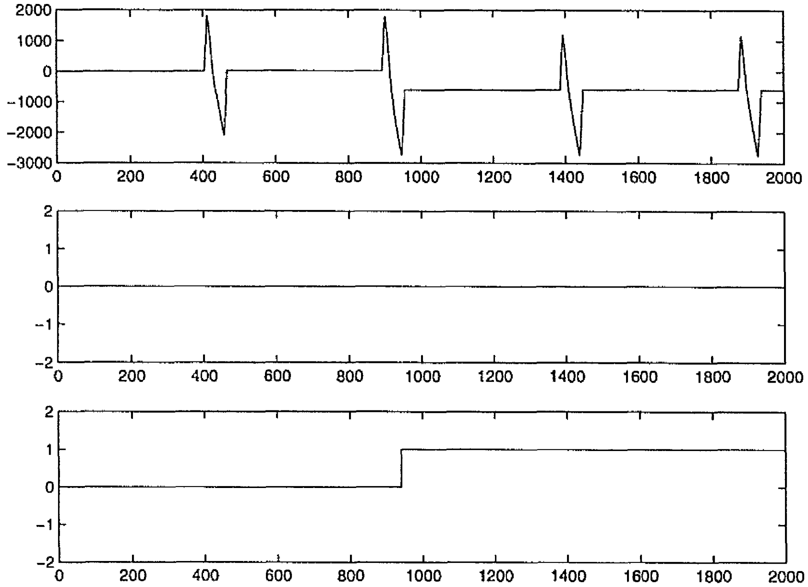


Figure 34: Tank 8 Volume (litres) Prediction & Measurements (Dashed)



Figures 35-37: Tank 8 Input To Tool 6b, Positive & Negative Cusum Outputs

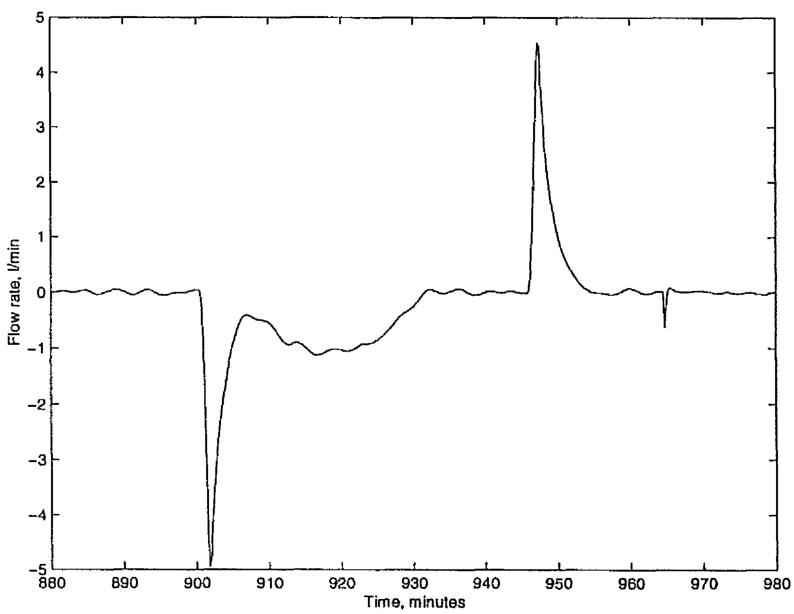


Figure 38: Observed flow rate to hidden inventory

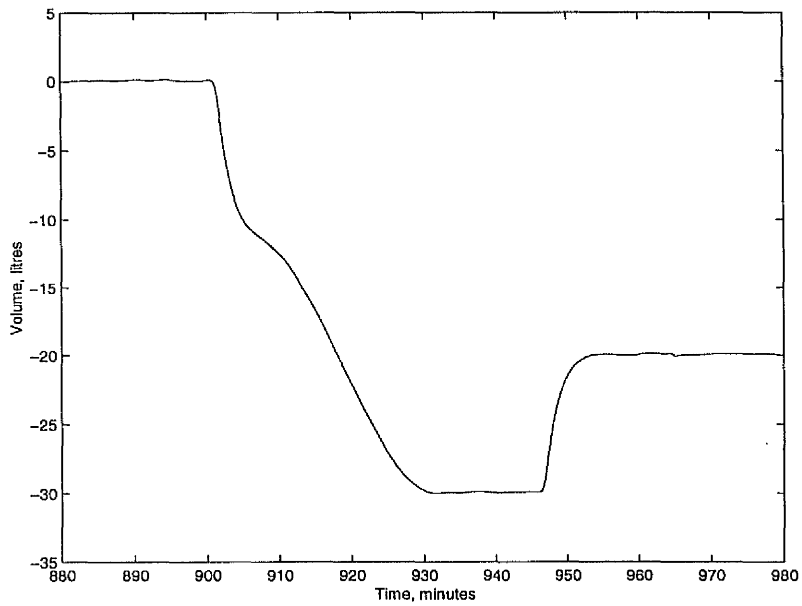


Figure 39: Integral of flow rate to hidden inventory

A.6.5 Case 4: Slower diversion from buffer tank 9

Case 4 describes the procedures that would ensue if 0.02 l/min of 30 g/l Pu concentration solution was to be continually 'taken' from the buffer tank after Cycle 2. Depending on the tolerances specified, a disagreement would arise in Tool 7b-1 within a few days (Table 9). Tool7b-2 would be then look at the redistribution of these errors and identify the problem to the correct buffer tank (Table 10).

Tool 7b-1

	1 day(0-1500)	2 days (0-2600)	3 days (0-4400)	4 days (0-5500)
Tank 1	0.17	0.25	0.50	0.48
Tank 2	-9.26	-9.10	-9.14	-9.22
Tank 3	-0.19	-0.09	-0.38	-0.36
Tank 4	-6.70	-6.62	-7.25	-7.55
Solex 1	-0.00	-0.00	-0.01	-0.00
Tank 5	0.47	1.44	-1.87	-0.93
Tank 6	-0.39	0.51	-0.31	-0.48
Tank 7	-8.60	-8.61	-7.94	-8.10
Solex 2	0.11	0.06	-0.02	-0.18
Tank 8	49.00	93.79	154.93	163.48
Tank 9	497.15	868.45	1442.33	1819.11
Tank 10	93.04	215.09	374.42	473.31
Concentrator	-0.15	0.01	-0.40	5.76
Tank 11	1.98	12.37	20.63	25.18
Tank 12	-47.84	-20.38	-85.78	-22.29

Table 9: Plutonium mass disagreements: simulated.vs.measured

Tool 7b-2

	Error	Redistribution
Tank 1	0.48	0.48
Tank 2	-9.22	-9.22
Tank 3	-0.36	-0.36
Tank 4	-7.55	-7.55
Solex 1	-0.00	-0.00
Tank 5	-0.93	-0.93
Tank 6	-0.48	4.72
Tank 7	-8.10	50.00
Solex 2	-0.18	50.00
Tank 8	163.48	50.00
Tank 9	1819.11	2242.24
Tank 10	473.31	50.00
Concentrator	5.76	5.76
Tank 11	25.18	25.18
Tank 12	-22.29	-22.29

Table 10: Tank 9 constant diversion

A.6.6 Case 5: Slower diversion from the inlet of Tank 8

Case 4 describes the procedures that would ensue if 0.02 l/min of 30 g/l Pu concentration solution was to be continually 'taken' from the Cycle 2 outlet. Depending on the tolerances specified, a disagreement would arise in Tool 7b-1 within a few days. At this point samples would have to be analysed to locate the area of concern.

Tool 7b

Table 11 shows the build-up in error in Tank 12 over four days. The build-up in Tank 12 would trigger the collection of samples (Table 12), which would be compared with the simulation results. This would point to the problem lying in the vicinity of the Cycle 2 feed tank, Cycle 2 itself and the Cycle 2 receiving tank. Redistribution (Table 13) would then be applied on the basis that the problem is with the mid-unit (i.e. the cycle itself). Appropriate flow rate correction terms would be generated and applied to the real-time database and an event would be generated that would point to the general vicinity.

	1 day(0-1500)	2 days (0-2600)	3 days (0-4400)	4 days (0-5500)
Tank 1	0.24	0.30	0.52	0.72
Tank 2	-8.51	-8.61	-8.58	-8.52
Tank 3	-0.21	-0.23	-0.23	-0.54
Tank 4	-6.57	-6.65	-7.18	-7.39
Solex 1	-0.01	0.00	-0.01	-0.00
Tank 5	0.59	1.18	-1.91	-1.45
Tank 6	-1.00	0.22	-0.55	-0.55
Tank 7	-8.46	-7.87	-7.29	-8.05
Solex 2	0.02	-0.00	-0.06	-0.24
Tank 8	-27.38	-42.92	-109.76	-153.40
Tank 9	2.08	1.11	1.78	2.80
Tank 10	11.33	78.88	129.04	173.69
Concentrator	-1.70	-3.54	-1.35	5.09
Tank 11	11.45	6.33	19.98	18.75
Tank 12	130.08	512.59	1180.83	1852.60

Table 11: Diversion tank 8

	Sample	Simulated
Tank 3	2.00	2.00
Tank 6	7.00	6.9992
Tank 9	31.45	31.98
Tank 12	221.78	226.61

Table 12: Sample data

	Error	Redistribution
Tank 1	0.72	0.72
Tank 2	-8.52	-8.52
Tank 3	-0.54	-0.54
Tank 4	-7.39	-7.39
Solex 1	-0.00	-0.00
Tank 5	-1.45	-1.45
Tank 6	-0.55	-0.55
Tank 7	-8.05	-8.05
Solex 2	-0.24	1599.29
Tank 8	-153.40	50.00
Tank 9	2.80	50.00
Tank 10	173.69	50.00
Concentrator	5.09	50.00
Tank 11	18.75	50.00
Tank 12	1852.60	50.00

Table 13: Redistribution after 4 days on the basis of the sample data

A.6.7 Case 6: Substitution Of Solution With Acid

A relatively large quantity of solution, (50 litres at 31.0 g/litre = 1550 gms of Pu) is replaced in the buffer tank upstream of the concentrator with the same quantity of nitric acid and of the same density (50 litres of 2.32 molarity). In theory if the switch is made 'cleanly' no change in either level or density should be observed in the tank. In practice the chance of achieving this must be very small so some variation is likely to be observed. There are probably many different ways in which the buffer tank level & density might vary so here we assume the worst case: that it is not obvious that it is not just process noise in which case considerable reliance would have to be placed on corroborating evidence. Although there would be various short-term effects on the units downstream, these would not be detectable , so detection would be based on observing the medium-term effects through the plutonium balance (Tool 7b). Table 14 shows the results obtained from Tool 7b over the day of the incident plus the following day. The incident took place in the early afternoon. Note that the error in tank 12 takes time to build-up as material (or lack of it) has to travel through the rest of the plant.

	End of day 1	End of day 2
Tank 1	0.19	0.39
Tank 2	-9.40	-9.37
Tank 3	-0.04	-0.09
Tank 4	-3.94	-4.26
Solex 1	-0.01	-0.01
Tank 5	-0.04	2.71
Tank 6	-0.60	-0.19
Tank 7	-6.58	-7.75
Solex 2	-0.23	-0.33
Tank 8	-32.87	-50.86
Tank 9	3.71	0.22
Tank 10	-2.82	7.45
Concentrator	7.21	0.27
Tank 11	4.82	9.06
Tank 12	970.50	1539.33

Table 14: Tool 7b Errors

<u>Tool 7b</u>

The build-up in Tank 12 would trigger the collection of samples, which would be compared with the simulation results. Depending on timing, this might provide evidence that a substitution has occurred, and then again, it might not. Either way the problem would be located as lying in the vicinity of the Concentrator feed tank, the Concentrator itself and the Concentrator receiving tank. Redistribution would then be applied on the basis that the problem is with the mid-unit (i.e. the Concentrator). Appropriate flow rate correction terms would be generated and applied to the real-time database and an event would be generated that would point to the general vicinity.

APPENDIX 7: SYSTEMATIC MULTIPLICATIVE BIASES IN TANKS

A number of test cases were examined to show the effect of multiplicative errors and how Tool7b-4 locates them. A plant simulation was first executed to produce a set of *real plant data* including the *real tank volume* and the *real plutonium inventory* for each tank. A set of multiplicative errors was now hypothesised and applied to this data to produce a set of *plant data*. This *plant data* was then analysed to estimate the flow rates through the plant, which was input into the plant simulation to predict *tank volumes* and *plutonium inventories*.

Table 15 indicates which tank volumes were used in the estimation of the various flow rates. The differences between the *real tank volumes* and the *tank volumes*, and between the Tank 12 *real plutonium inventory* and the *plutonium inventory*, were tabulated at the end of each day. This was repeated for a number of different of error sets (Tables 16-24). The differences were now analysed by Tool 7b-4 on the basis of one set of daily measurements. Some of the results are given in Tables 25-28. Finally a test case was performed that involved more than one set of measurements and the results are given in Table A7.14.

The following test cases were generated:

a) the clean (i.e. unbiased) case (Table 16)
b) +1% on Tank 8 only (Table 17)
c) +1% on Tank 9 only (Table 18)
d) +1% on Tank 10 only (Table 19)
e) +1% on Tank 3 only (Table 20)
f) Tank 8: +5%, Tanks 9 & 10: –5% only (Table 21)
g) +1% on the 1st tank, -1% on the 2nd, and so on (Tables 22 & 23)
h) +1 % on all the tanks (Table 24)

Notes:
1. Tank 12 is the product accountancy tank.
2. The plutonium inventories in the solvent extraction cycles and in the concentrator are maintained constant.
3. There is perfect sampling (i.e. all the solution is returned) and no evaporation.

Case a)

A no bias case was performed to give an indication of the magnitude of the disagreements that arise when only random errors are imposed.

Cases b) to e)

These involved only one bias. In each case Tool 7b-4's first hypothesis located the bias, quantification at the end of day 2 (Table 25) resulted in estimates, which when substituted back into the simulation eliminated the disagreements. The revised disagreements were: in case b) 1.02 (Tank 8), in case c) 0.01 l (Tank 8) & -0.22 l (Tank 10), in case d) 2.04 l (Tank 10) and in case e) 50.79 l (Tank 3) & 30.33 l (Tank 4). Note the upstream tanks (Tanks 3&4) have larger acceptable disagreements because the tanks are much larger.

Case f)

A single bias hypothesis (in Tank 8) lead to an initial estimate of 9.4%, which, on re-running the simulation, resulted in a Tank 8 error of 21.9 l and a Tank 12 Pu inventory error of –416.6 gm. The hypothesis involving all three biases (in the second tank-set) was evaluated by re-running the simulation with different error distributions until the Pu concentration in Tank 12 agreed with that 'measured' (224.0). Some of the results are shown in Table 26. It can be seen that the output would be {5, -5, -5} if the test for convergence was to be based on Pu concentration , and would be about {6, -4, -4} if it was to be based on the Pu inventory. In practice it would depend on what is deemed to be more accurate.

Case g)

A similar approach would be adopted here. Unfortunately the software used to perform these evaluations could only be executed semi-automatically making it extremely time-consuming to produce a converged solution. Tables 27 and 28 gives the first two iterations, which was all that was generated.

Case h)

This has been included to show what might be considered to be the hardest case. The build-up in Tanks 1 & 12 indicate that there is a problem, other than this there is little information. It is important to understand that the analyses described can be performed regularly. Thus, for instance, a hypothesis can be made, which correlates with the day's data, then tested against data collected on subsequent days. It is difficult to 'pre-plan' a strategy for dealing with this eventuality until the actual data collection process starts.

	Flow in	**Flow out**
Tank 1	Input Acc. Tank	Tank 2
Tank 2	Tank 2	Tank 2
Tank 3	Tank 2	Tank 3
Tank 4	Tank 3	Tank 4
Tank 5	Tank 5	Tank 6
Tank 6	Tank 6	Tank 6
Tank 7	Tank 6	Tank 7
Tank 8	Tank 8	Tank 9
Tank 9	Tank 9	Tank 9
Tank 10	Tank 9	Tank 10
Tank 11	Tank 11	Tank 11
Tank 12	Tank 11	Tank 12

Table 15 Identifies the tank volume used to estimate the flow rate

Tank	**End of Day 2**
Tank 1	-4.87
Tank 2	-5.51
Tank 3	-5.56
Tank 4	-3.35
Tank 5	0.06
Tank 6	-1.06
Tank 7	-1.50
Tank 8	-1.31
Tank 9	-0.20
Tank 10	2.27
Tank 11	-0.00
Tank 12	-2.35
Pu(Tank 12)	-105.86

Table 16 No biases (i.e. the base case)

Tank	**End Of Day 1**	**End Of Day 2**
Tank 1	0.21	0.24
Tank 2	-4.67	-4.64
Tank 3	-0.16	-0.18
Tank 4	-2.81	-2.89
Tank 5	0.12	0.61
Tank 6	-0.10	-0.23
Tank 7	-0.94	-1.23
Tank 8	13.60	23.26
Tank 9	0.05	0.04
Tank 10	0.08	0.47
Tank 11	-0.02	-0.03
Tank 12	-1.61	-2.20
Pu(Tank 12)	-132.92	-307.69

Table 17 +1% on Tank 8

Tank	**End Of Day 1**	**End Of Day 2**
Tank 1	0.21	0.24
Tank 2	-4.67	-4.64
Tank 3	-0.16	-0.18
Tank 4	-2.60	-2.47
Tank 5	0.37	0.48
Tank 6	-0.10	-0.23
Tank 7	4.64	4.64
Tank 8	-16.51	-26.53
Tank 9	0.13	0.12
Tank 10	16.04	26.57
Tank 11	-0.02	-0.03
Tank 12	-1.60	-2.17
Pu(Tank 12)	-195.32	-148.45

Table 18 +1% on Tank 9

Tank	**End Of Day 1**	**End Of Day 2**
Tank 1	0.21	0.24
Tank 2	-4.67	-4.64
Tank 3	-0.16	-0.18
Tank 4	-4.00	-3.41
Tank 5	-0.15	0.13
Tank 6	-0.10	-0.23
Tank 7	-0.59	-0.57
Tank 8	-1.96	-1.90
Tank 9	0.05	0.04
Tank 10	-14.37	-23.37
Tank 11	-0.02	-0.03
Tank 12	-1.61	-2.19
Pu(Tank 12)	408.25	729.46

Table 19 +1% on Tank 10

Tank	**End Of Day 1**	**End Of Day 2**
Tank 1	0.21	0.24
Tank 2	-4.67	-4.64
Tank 3	-231.74	-463.36
Tank 4	230.00	383.13
Tank 5	0.54	0.54
Tank 6	-0.10	-0.23
Tank 7	-1.26	-1.43
Tank 8	-1.35	-1.15
Tank 9	0.05	0.04
Tank 10	1.77	2.49
Tank 11	-0.02	-0.02
Tank 12	-1.60	-2.17
Pu(Tank 12)	-138.96	-90.94

Table 20 +1% on Tank 3

Tank	**End Of Day 1**	**End Of Day 2**
Tank 1	0.09	0.14
Tank 2	-4.59	-4.66
Tank 3	-0.12	-0.28
Tank 4	-3.68	-3.80
Tank 5	0.24	0.43
Tank 6	-0.26	-0.25
Tank 7	-0.44	-0.66
Tank 8	145.42	242.78
Tank 9	0.15	-0.02
Tank 10	0.93	1.82
Tank 11	-0.02	-0.05
Tank 12	-1.58	-2.14
Pu(Tank 12)	-3387.63	

Table 21 +5% on Tank 8, -5% on Tanks 9 & 10

Tank	**End Of Day 1**	**End Of Day 2**
Tank 1	231.64	463.40
Tank 2	-4.71	-4.75
Tank 3	-463.28	-926.51
Tank 4	460.66	769.43
Tank 5	131.85	220.21
Tank 6	-0.01	-0.00
Tank 7	-133.65	-221.80
Tank 8	-30.25	-50.92
Tank 9	0.17	0.30
Tank 10	29.86	51.33
Tank 11	-0.03	-0.03
Tank 12	0.82	1.17
Pu(Tank 12)	1763.94	2201.14

Table 22 +1%, followed by –1% down the plant

Tank	Bias	Flow in	Flow out	Flow in	Flow out	Inventory
1	+1	Accountancy	Tank 2	1.0	0.99	↑
2	-1	Tank 2	Tank 2	0.99	0.99	-
3	+1	Tank 2	Tank 3	0.99	1.01	↓
4	-1	Tank 3	Tank 4	1.01	0.99	↑
Solex 1	------	Tank 4	Tank 5	0.99	1.01	Pu g/l decrease
5	+1	Tank 5	Tank 6	1.01	0.99	↑
6	-1	Tank 6	Tank 6	0.99	0.99	-
7	+1	Tank 6	Tank 7	0.99	1.01	↓
Solex 2	------	Tank 7	Tank 8	1.01	0.99	Pu g/l increase
8	-1	Tank 8	Tank 9	0.99	1.01	↓
9	+1	Tank 9	Tank 9	1.01	1.01	-
10	-1	Tank 9	Tank 10	1.01	0.99	↑
Concentrator	------	Tank 10	Tank 11	0.99	1.01	Pu g/l decrease
11	+1	Tank 11	Tank 11	1.01	1.01	-
12	-1	Tank 11	Accountancy	1.01	1.00	↑

Table 23 Predicted results for Pu inventory for +1%,-1% on plant

Tank	End Of Day 1	End Of Day 2
Tank 1	-231.50	-463.03
Tank 2	-4.81	-4.85
Tank 3	-0.14	-0.08
Tank 4	-2.45	-2.69
Tank 5	-0.45	-0.33
Tank 6	-0.01	-0.00
Tank 7	-1.38	-1.27
Tank 8	-0.93	-2.02
Tank 9	0.17	0.30
Tank 10	1.80	3.85
Tank 11	-0.03	-0.03
Tank 12	0.83	1.17
Pu(Tank 12)	408.40	661.02

Table 24 +1% throughout plant

	Tank 8	Tank 9	Tank 10	Tank 3	Alternate down plant	Tanks 8, 9 & 10
	+1	+1	+1	+1	+1, –1	+5 , -5, -5
Tank 1	0	0	0	0	0	0
Tank 2	0	0	0	0	-1.3500	0
Tank 3	0	0	0	0.9100	0	0
Tank 4	0	0	0	0	-1.9700	0
Tank 5	0	0	0	0	0	0
Tank 6	0	0	0	0	-2.0200	0
Tank 7	0	0	0	0	0	0
Tank 8	0.9400	0	0	0	-2.0600	9.400
Tank 9	0	1.0836	0	0	0	0
Tank 10	0	0	0.9000	0	-1.9600	0
Tank 11	0	0	0	0	0	0
Tank 12	0	0	0	0	0	0

Table 25 First attempt

Errors	{9.4, 0, 0}	{0, -10, -10}	{5, -5, -5}
Tank 8 volume	21.9	-15.6	-2.1
Tank 9 volume	-0.27	-0.29	-0.28
Tank 10 volume	1.57	1.74	1.66
Tank 12 Pu inv	-417	783	373
Tank-set 3 Pu conc.	-1.7	1.2	-0.4
Tank 12 Pu conc.	2.0	-1.1	0.0

Table 26 Results obtained for various Tank-set 3 bias distributions when biases are {5, -5, -5}

Tank	Bias Estimate	Volume Error	Pu error
1	0.0	-169.1147	-328.2697
2	-1.35	-5.5425	-9.6220
3	0.0	-304.4695	-597.9513
4	-1.97	1.3926	3.4211
5	0.0	-4.8161	-36.3351
6	-2.02	-1.1979	-0.3524
7	0.0	3.0166	24.4180
8	-2.06	0.0643	-14.7695
9	0.0	0.0347	9.5198
10	-1.96	2.8245	90.7659
11	0.0	-0.0323	-7.1908
12	0.0	1.1519	603.9800

Tank set	Pu Actual	Pu Estimate
1 2 3 4	2	2
5 6 7	7	6.998
8 9 10	31.5	31.47
11 12	224	222.89

Table 27 Evaluation of 1st hypothesis for case g)

Tank	Bias Estimate	Volume Error	Pu error
1	0.0	-169.11	-328.27
2	-1.35	-5.54	-9.62
3	0.0	-304.47	-597.95
4	-2.01	-14.52	-28.39
5	2.06	-4.54	-34.68
6	0.0	-1.17	0.04
7	0.91	-121.57	-867.54
8	-2.14	2.09	49.40
9	0.0	0.03	9.19
10	-1.90	4.34	140.37
11	0.0	-0.03	-7.27
12	0.0	1.15	1410.34

Tank set	Pu Actual	Pu Estimate
1 2 3 4	2	2
5 6 7	7	7.15
8 9 10	31.5	31.816
11 12	224	225.195

Table 28 Evaluation of 2nd hypothesis for case g)

APPENDIX 8: SOME EXAMPLES OF THE DATA OBJECTS CREATED BY SOME OF THE TOOLS

Case 1: Diversion Not During Export

$t_{start} = 1500$
$t_{finish} = 2000$

Tool 6a-1:

Create a sub-event with name Sub-event_id1:

    diagnoses = (diagnosis_id1)
    scores = (1)
    symptoms = (symptom_id1)

    diagnosis_id1::sd_id1 is a sub-class:
        path-type = measurement_error
        path-id = V04
        quantity = time profile from 1600 to 1700 (2 columns t,V)

    symptom_id1 is a sub-class:
        error-in = flow
        path-id = c**t04
        start-time = 1600
        stop-time = 1700

Create a sub-event with name Sub-event_id2:

    diagnoses = (diagnosis_id1)
    scores = (1)
    symptoms = (symptom_id1)

    diagnosis_id1::sd_id1 is a sub-class:
        path-type = measurement_error
        path-id = V04
        quantity = time profile from 1800 to 1950 (2 columns t,V)

    symptom_id1 is a sub-class:
        error-in = flow
        path-id = c**t04
        start-time = 1800
        stop-time = 1950

Tool 7a-2 followed by Tool 6a-2:

Add to sub-event Sub-event_id:
diagnoses = diagnosis_id1, diagnosis_id2, diagnosis_id3
scores = (1 2 3)

with sub-class diagnosis_id2::sd_id1:
path-type = molarity
path-id = c**h**
quantity = time profile of $[H^+]_{out}$ over the period 1600 - 1700

and sub-class diagnosis_id2::sd_id2:
path-type = Pu
path-id = c**h**
quantity = time profile of $[Pu]_{out}$ over the period 1600 - 1700

and sub-class diagnosis_id3::sd_id1:
path-type = flow
path-id = c**h**
quantity = time profile of flow rate to hidden inventory

Tool 7c-1:

sub-event_id1::link-with = sub-event_id2
sub-event_id1::link-type = 'primary_effect'
sub-event_id2::link-with = sub-event_id1
sub-event_id2::link-type = 'secondary_effect'
sub-event_id1::diagnosis = diagnosis_id3
sub-event_id1::score = 2

Make-event event_id:

event_description = '*Solex_hinv*'
Sub-events = (sub-event_id1)

Case 3: Diversion During Export

$t_{start} = 1700$
$t_{finish} = 2200$

Tool 6b-1:

Create a sub-event with name Sub-event_id1:
    diagnoses = (diagnosis_id1)
    scores = (1)
    symptoms = (symptom_id1)

    diagnosis_id1::sd_id1 is a sub-class:
        path-type = measurement_error
        path-id = V05
        quantity = time profile 1890 to 2200 (2 columns t, V)

    symptom_id is a sub-class:
        error-in = level
        path_id = V05
        start-time = 1890
        stop-time = 2000

Tool 7a-2 followed by Tool 6a-2:

Add to sub-event Sub-event_id:
    diagnoses = (diagnosis_id1, diagnosis_id2)

  with sub-class diagnosis_id2::sd_id2:
        path-type = flow
        path-id = t04h04
        quantity = time profile 1890 to 2200 (2 columns t, V)

Tool 7c-1:

sub-event_id1::diagnosis = diagnosis_id2
sub-event_id1::score = 3

Make-event event_id:

  event_description = '*Tank_hinv*'
  Sub-events = (sub-event_id)